Optimizing the Energy-Latency Trade-Off in NB-IoT with PSM and eDRX

Reference:
Sultania Ashish Kumar, Blondia Christian, Famaey Jeroen.- Optimizing the Energy-Latency Trade-Off in NB-IoT with PSM and eDRX
IEEE internet of things journal - ISSN 2327-4662 - 8:15(2021), p. 12436-12454
Full text (Publisher's DOI): https://doi.org/10.1109/JIOT.2021.3063435
To cite this reference: https://hdl.handle.net/10067/1760070151162165141

Optimizing the Energy-Latency Trade-Off in NB-IoT with PSM and eDRX

Ashish Kumar Sultania, Chris Blondia, *Senior Member, IEEE* and Jeroen Famaey, *Senior Member, IEEE*

***Abstract*—Narrowband Internet of Things (NB-IoT) is becoming one of the most promising low power wide area (LPWA) networking technologies. It can support more than 50 000 devices within a cell using licensed spectrum. NB-IoT provides low energy consumption, reliable connectivity and deep indoor coverage for the device, making it a good candidate for IoT use cases. NB-IoT introduces two novel energy-saving techniques, namely extended discontinuous reception (eDRX) and power saving mode (PSM). This paper presents a Markov chain model to evaluate the power consumption and latency of NB-IoT devices using PSM and eDRX. By exploiting the characteristics of the steady-state distribution of the Markov chain, the probabilities in steady-state can be obtained explicitly. Based on these probabilities, we calculate the system downlink (DL) latency as a function of different timers of these power-saving features. We also compare the model to simulation results obtained from the ns-3 event-based network simulator, to determine its accuracy. The results show that its performance in terms of energy and latency is comparable. Our model is accurate with consideration of the protocol details and the new Radio Resource Control (RRC) Idle features of NB-IoT. The results show that the analytical model achieves an average accuracy of more than 91% for power consumption and DL latency. Lastly, we use the model to automatically determine the optimal parameter set in terms of latency and power consumption for various IoT use cases with different traffic requirements, based on multi-objective analysis of the Pareto front.**

***Index Terms*—NB-IoT, power consumption, power saving mode, extended discontinuous reception, simulation, analytical model, Markov chain.**

I. Introduction

THE cellular IoT (CIoT) market is projected to reach 5.1 billion USD by 2025, with a growth rate of 12.3% [1]. The CIoT is a low power wide area (LPWA) network standard enabling a wide range of low powered devices using cellular telecommunications bands to connect to the Internet. It offers reliable connectivity, excellent coverage, massive capacity and low-cost hardware. Generally, cellular networks such as Long-Term Evolution (LTE) and Wideband Code Division Multiple Access (WCDMA) are unable to deliver all these benefits. Therefore, the 3rd Generation Partnership Project (3GPP) has proposed Narrowband Internet of Things (NB-IoT) as the main LTE based CIoT standard. NB-IoT is designed to support a wide range of IoT applications such as smart parking, smart cities, smart agriculture, industrial monitoring, and smoke detectors.

IoT devices are generally battery powered and should be able to operate for several years without battery replacement. The existing schemes used for power saving in LTE are unable to deliver such long lifetimes. Therefore, NB-IoT proposes two novel power-saving schemes: Power Saving Mode (PSM) and Extended Discontinuous Reception (eDRX) [2]. PSM allows devices to enter into a deep sleep mode by switching off most of their circuitry while staying registered to the network. In this mode, the device is not reachable from the network. However, it can wake up at any time to transmit data towards the network. As the device can wake up at any time to send the uplink (UL) data, it has low UL latency. However, as it can only receive Downlink (DL) data once it leaves PSM mode (due to an UL transmission or expiration of the PSM timer), its DL latency can be extremely high. As such, PSM is mostly suited for IoT use cases with sporadic UL transmissions with DL immediately after UL or not at all. In eDRX, devices enter into a sleep mode where they do not listen to the radio channel for a defined period and become active periodically to receive a paging message from the network for possible incoming data. As such, eDRX provides a trade-off between power consumption and DL latency, as it still allows periodically listening for DL data. The eDRX and PSM modes can also be used jointly, where the device first performs eDRX for a period of time and then goes into PSM. The user equipment (UE) is free to configure the different eDRX and PSM parameters. The network may accept these parameters' values or negotiate different ones. However, different configurations can lead to vastly different power consumption and DL latency, and the optimal parameters thus depend on the considered IoT use case. For example, the use cases for which DL latency is the key metric should have a long eDRX length with short paging cycle.

In this article, we aim to evaluate the power consumption and DL latency of NB-IoT devices for different PSM and eDRX configurations, as well as determine the optimal configuration for specific IoT use cases. In general, there are three techniques for performance evaluation of a system a) real-time active system measurements b) mathematical analysis and c) simulation. All these techniques have their strengths and weaknesses. In this paper, we present the mathematical and the simulated analysis of the NB-IoT power-saving schemes. Generally, researchers use network simulation for its flexibility to evaluate a wide range of scenarios , and it is high accuracy and fidelity. For many years, the ns-3 network simulation tool has been the de-facto standard in academic research for evaluating networking protocols, and therefore, we chose the ns-3 simulator to verify our mathematical model.

The main contributions of this paper are:

- The first-ever NB-IoT performance model, based on

Ashish Kumar Sultania, Chris Blondia and Jeroen Famaey are with University of Antwerp and imec, Belgium (e-mail: ashishkumar.sultania, chris.blondia, jeroen.famaey @uantwerpen.be)

Markov chains that combines both PSM and eDRX for both UL and DL transmissions, evaluating the trade-off between power consumption and DL latency.
- Evaluation of the Markov model's accuracy by comparing it to our implementation of PSM and eDRX in the ns-3 network simulator.
- Performance characterization and optimization of PSM and eDRX parameters for different IoT use cases based on a Pareto front analysis obtained via the Markov model.

The rest of this paper is organized as follows. Section II focuses on related work. Section III describes the overview of NB-IoT. Section IV then presents the analytical model based on the Markov chain. Section V looks at the results, validation and optimal eDRX and PSM parameters' values. Finally, Section VI concludes this paper.

II. Related Work

NB-IoT has been commercially recognized as one of the most promising LPWA technologies. Many operators like Orange, Vodafone, and TIM are already offering commercial NB-IoT services in many countries. However, still, the majority of research analysis are performed considering only UL traffic. This might be because most of the IoT use cases consider sensor-based monitoring, where the majority of traffic is expected to be UL. The UL latency generally consists of broadcast latency, random access (RA) latency, and data transmission latency. Maldonado et al. [3] evaluated the power consumption of an NB-IoT device considering different coverage levels and different UL Interpacket Arrival Times (IATs) by reducing signaling using the mechanisms known as Control Plane Cellular IoT (CP) optimization and User Plane Cellular IoT (UP) optimization. Recently, they also defined an analytical model based on Markov chains to calculate the power consumption of UEs [4]. Their model estimates the power consumption and latency only for a device sending periodic UL data using CP optimization. In 2017, Lee at al. [5] proposed a prediction-based energy-saving mechanism for UL transmission. They allocate the resources in advance based on response time for each previous transmission and achieve up to 34 % of battery saving.

Bello et al. [6] developed a semi-Markov chain to evaluate power consumption and delay performance under periodic UL data transmission. Furthermore, they also introduced an optimization model of the PSM timers that minimizes energy consumption and the average delay. In our previous work [7], we also proposed a preliminary analytical model to analyze the average energy consumption of NB-IoT devices using both PSM and eDRX. However, that work focused on UL transmissions with a Poisson arrival process. Liu et al. [8] propose a Markov chain to analyze PSM with DL and UL traffic. They also use a genetic algorithm to obtain the power saving factor that is the fraction of time the UE spends in the PSM mode. However, the paper does not study the trade-off of DL latency and power consumption. Other works such as [9], [10], and [11] solely focused on performance models to maximize the Random Access Channel (RACH) success probability concerning data transmission and RA latency.

Nevertheless, DL data transmission is also important in some use cases such as over-the-air device configuration, control loops, voice calls, and polling-based data retrieval. In these scenarios, to receive DL data or notifications, the device needs to continuously monitor the DL channel, which reduces the battery life. Oh et al. [12] modeled and evaluated the battery consumption rate for DL data reception only. This research lacks the analysis of UL transmissions and latency. Several works have evaluated the effect of eDRX parameters on system performance [13] [14].

An initial work on the development of an NB-IoT simulation platform based on OPNET is presented in [15] and validated for the low-rate data transmission of NB-IoT, focusing on its physical layer characteristics. Soussi et al. [16] presented an alternative implementation of the NB-IoT physical layer in ns-3, further building upon the ns-3 LTE module. We extended this work with support for various data link layer timers and features, such as the Radio Resource Control (RRC) connection and idle state, as well as eDRX and PSM [17]. Furthermore, Lauridsen et al. [18] presented empirical power consumption measurements of two NB-IoT UEs.

However, the works mentioned above have not provided a complete analysis of IoT applications with both UL and DL transmissions simultaneously. Also, the energy-latency trade-off when using NB-IoT with eDRX and PSM has not been studied. In contrast, this paper presents the complete picture of the NB-IoT energy saving schemes.

III. NB-IoT: Overview

This section gives an overview of the NB-IoT device network access procedure and its power saving schemes.

A. General overview of NB-IoT

When the device is powered on, it fetches the frequency, synchronizing timings, detects signal quality, and some important configurations from the network. It decides to camp-on to the best cell and starts the RACH process. The RACH helps the device to obtain the resources for the RRC connection request. This RRC connection setup is an important step because only after that, the device and the network exchange data. When the RRC connection is established, the device is said to be in RRC Connected state. In NB-IoT, there are two RRC states for devices, namely, *RRC Connected* and *RRC Idle* as shown in Figure 1. When the device releases its active RRC connection, it moves to the RRC Idle state. The device in RRC Connected state consumes more energy, as it gets dedicated bearers established to begin the data transmission and needs to monitor the DL channel in all the subframes (SFs) except for the SFs for UL transmission. The control channel it monitors is called the Narrowband Physical Downlink Control Channel (NPDCCH), which is required to receive the DL data notification or UL data grant from the eNodeB (eNB). These notifiers are known as paging indicators, and the procedure of indicating to the device that data is available is called paging. The time instance of paging is known as a paging occasion (PO). The device receives the data from the network over the Narrowband Physical Downlink Shared Channel (NPDSCH)

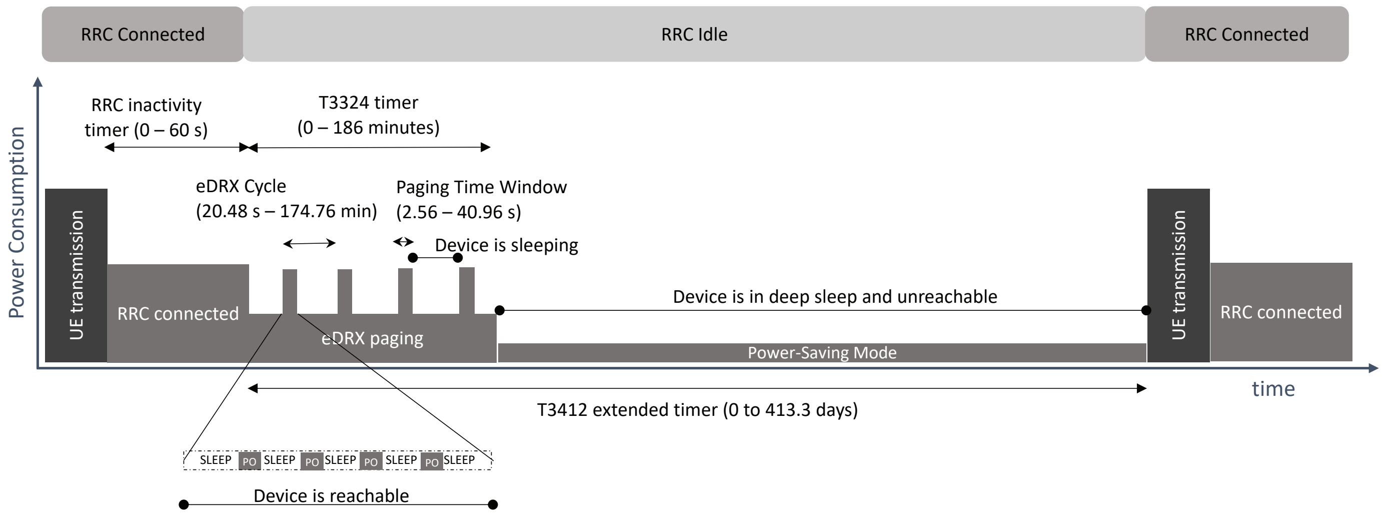


Fig. 1: Overview of the NB-IoT UE duty cycle states

and transmits over the Narrowband Physical Uplink Shared Channel (NPUSCH). There is a network defined timer known as *RRC Inactivity Timer*, whose expiration makes the device transition from the RRC Connected to RRC Idle state. The eNB initiates the RRC Release message on the expiration of the RRC Inactivity timer. Instead of waiting for the RRC release message from the network, the UE may also use Release Assistance Indication (RAI) to indicate that no further uplink or downlink data transmissions are expected. This can help the network to decide if the connection can be released, and thus reduces the period the UE spends in the RRC connected state. Hence, it conserves even more energy. However, the RAI feature has been introduced in Release 14 and is not supported by all the NB-IoT device categories. Therefore, we omitted its study from this work.

In the RRC Idle state, NB-IoT defines two power-saving schemes, i.e., eDRX and PSM. In this state, the device saves a lot of energy as it stops continuously monitoring the DL channels. This, however, has a significant impact on the latency of DL transmissions, as POs are either less frequent (in eDRX) or non-existent (in PSM). As such, DL data is buffered by the NB-IoT network until the next PO occurs. These power-saving schemes are described in the remainder of this section.

B. RRC Idle state

In RRC idle state, the UE does not have an established physical connection to the eNB. However, the network has its identity and the location based on the tracking area update (TAU). This section discusses the two power saving schemes in this state.

1) Extended discontinuous reception: It is similar to Discontinuous Reception (DRX) used in LTE systems but with longer timer values to achieve further improvement in energy consumption. The eDRX is designed to allow periodic DL data while minimizing energy consumption. The scheme works in cycles where each cycle consists of a short period "*On Duration*" during which the device monitors the DL control channel and a sleeping period during which the device saves its battery and stops monitoring the control channel. This feature can be used while the device is in either of the RRC states. In RRC connected state, it is named connected-mode DRX (C-DRX), and in the RRC Idle state, it is named idle-mode DRX (I-DRX). In IoT use cases, the RRC Connected period should be short, and therefore eDRX during RRC Idle makes a bigger contribution to the battery-saving compared to eDRX during RRC Connected. As such, in this paper, we have considered only eDRX during RRC Idle state, as shown in Figure 1. A timer T_{3324} (also known as Active timer) is defined as the eDRX state time of the device during which a device can still be periodically reachable by the network. For NB-IoT, it varies from 0 to 186 minutes [26]. During this period, the device monitors the channel for paging messages at the interval of the eDRX cycle, which can be configured up to 174.76 minutes [27]. An eDRX cycle consists of a Paging Time Window (PTW) time (between 2.56 and 40.96 seconds) followed by a sleep time [28]. The device monitors the channels during a few SFs, the Paging Occasions (PO) within the PTW. PTW thus involves cycles that alternate between periods of active listening and sleep. During paging, if the device receives a DL data notification from the eNB, it switches to the Connected state, and the eNB resets the RRC inactivity timer. If a DL packet arrives at the eNB in between paging events, the data is temporarily buffered by the network. This periodic DL reception during eDRX incurs additional latency. However, the UL latency is not affected, as the UE can directly move to the RRC Connected state as soon as UL data needs to be sent. As such, the UL latency depends on synchronization, broadcast information fetching, random access, resource allocation, data transmission, and feedback response delay [2]. Upon expiration of T_{3324} without any activity, the device can switch to PSM or restart the eDRX state if PSM is disabled.

2) Power saving mode: The main purpose of PSM is to minimize energy consumption while the device does not transmit or receive anything. The device is in a dormant state during the PSM and cannot receive DL data. It consumes

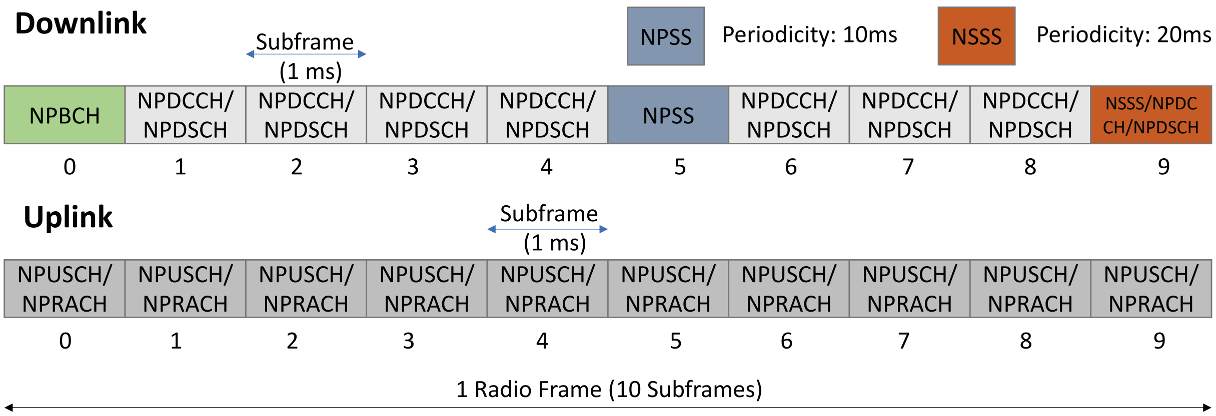


Fig. 2: NB-IoT uplink and downlink frame structure

energy similar to the switch-off state while it stays registered to the network. The PSM timer for NB-IoT is up to 413.3 days and is represented as $T_{3412}\ extended$ [26]. Upon expiration of T_{3412}, the UE monitors the channel for paging messages or performs a TAU for synchronization. Therefore, it can receive DL data only when PSM ends, which happens when the PSM timer expires or the UE switches to the Connected state (e.g., when it needs to send UL data). If a device in the PSM state generates a UL packet, it switches to the Paging state to monitor the control channel for the UL grant. The device will switch to the Connected state if it receives the grant and so the UL latency is only slightly affected. If the grant is rejected, the device switches back to the PSM state. All the incoming DL data during the PSM cycle is buffered by the network and sent to the device after PSM ends.

In comparison to PSM, eDRX saves less energy but provides better DL communication latency. Therefore, both the PSM and eDRX mechanisms can be used to adapt to different IoT scenarios. The state change depends on network traffic and device behavior. Hence, it becomes crucial to select an optimal eDRX configuration and paging period for various types of UL and DL traffic in order to balance the power-saving ratio and DL communication latency.

C. RRC Connected state

The UE switches to the RRC Connected state to transmit or receive some data. Therefore, this section presents an overview of both the DL and UL data transmission mechanism.

1) Downlink: The NB-IoT DL frame consists of three channels namely Narrowband Physical Broadcast Channel Narrowband (NPBCH), NPDCCH, and NPDSCH as presented in Figure 2. The DL frame also consists of three signals that are generated at the physical layer for synchronization and channel estimation functions. These signals are named as Narrowband Reference Signal (NRS), Narrowband Primary Synchronization Signal (NPSS), and Narrowband Secondary Synchronization Signal (NSSS). NPBCH carries the Narrowband Master Information Block (MIB-NB) at SF-0 and is transmitted over a time period of 640ms. MIB-NB carries some high-level information such as system timing and System Information Block (SIB1-NB) scheduling configurations. The UE uses the NPSS and the NSSS for time and frequency synchronization, and cell identity detection. The NPSS is transmitted at SF-5 of every radio frame, and NSSS at SF-9 of every even radio frame. The NRS is used to provide phase reference for the demodulation of the DL channels [2]. The NRS is transmitted in the SFs that carry NPBCH, NPDCCH and NPDSCH using 8 resource elements (REs). Other remaining SFs can be assigned to NPDCCH or NPDSCH. The NPDCCH indicates for which UE there is data in the NPDSCH, SFs containing NPDSCH and its repetition count. The NPDCCH carries the Downlink Control Information (DCI) which, depending on its functionality, has three different formats:

- DCI Format N0: It contains the information related to UL scheduling grants.
- DCI Format N1: It is used for NPDSCH and Narrowband Physical Random Access Channel (NPRACH) scheduling.
- DCI Format N2: It is used for paging and direct indication such as informing the UE about parameter modifications or issuing warning messages.

The DL scheduling information carries parameters such as modulation and coding scheme (MCS), SF assignment, NPDSCH repetition count (N_{Rep}^{DL}), and scheduling delay (k_{N1}). The MCS and the assigned number of SFs helps to select transport block size (TBS) using Table 16.4.1.5.1-1 mentioned in [21]. The TBS size varies from 2 to 85 bytes. For DL, NB-IoT uses only one modulation scheme that is quadrature phase-shift keying. After the reception of the DL data, the UE acknowledges it using Narrowband Physical Uplink Shared Channel (NPUSCH).

There are up to three coverage enhancement (CE) levels (0,1 and 2) to tackle with different radio conditions. To enhance the transmission reliability for different CE levels, the data and the associated control signaling have to be repeated several times. Additionally, in FDD, NB-IoT chooses half-duplex that means the UE can either receive or transmit, and can not performs both the operations simultaneously. Therefore, some guard SFs are needed in between every switch from transmission mode (Tx) to receive mode (Rx) or vice versa to provide the time to the UE to switch the radio activity. This scheduling delay is notified by the DCIs. Figure 3 mentions these delay

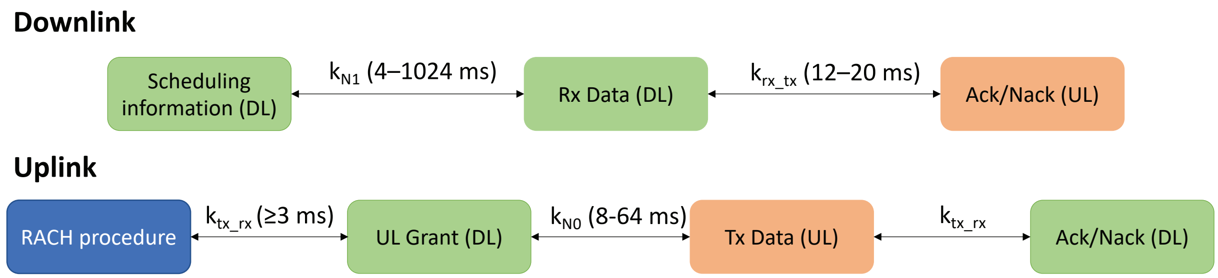


Fig. 3: Scheduling delay and data transmission in NB-IoT

values between the channels. There need to be at least 3 SFs gap when switching from Tx to Rx (k_{tx_rx}). The minimum scheduling delay between the end of Rx scheduling info and Rx data (k_{N1}) is 4 SFs. Whereas, the minimum scheduling delay increases to 8 ms (k_{N0}) and 12 ms (k_{rx_tx}) when the UE needs to switch from Rx to Tx. This is to allow it to decode the received data, switch the radio mode and prepare for the Tx. However, by introducing these scheduling delays, the data rate decreases.

2) Uplink: The NB-IoT UL frame consists of two channels namely NPRACH and NPUSCH. The NPRACH is used by the UE to access the network and to request radio resources to transmit its data. Figure 2 represents the time multiplexing of the NB-IoT UL physical channels which shows that both the channels can occupy any SFs. Therefore, except for the NPRACH transmission, the UE data can be sent over the NPUSCH. For NPRACH, based on the Cyclic Prefix (CP) length, two preamble formats are defined, format 0 and format 1. Each symbol group has a CP followed by 5 symbols. The length of five symbols is 1.333 ms. Each preamble is composed of 4 symbol groups transmitted without gaps which makes the complete preamble 5.6 and 6.4 ms for format 0 and 1, respectively [20]. The preamble format is broadcast in the system information. The frequency hopping is applied to the symbol group so that they can be transmitted on a different subcarrier. Depending on the different CE levels, the network can configure the NPRACH parameters such as NPRACH resource periodicity, number of preamble repetition count, number of subcarriers, and starting time of NPRACH resource.

The NPUSCH has two different formats, NPUSCH Format-1 (NPUSCH F1) and NPUSCH Format 2 (NPUSCH F2). NPUSCH F1 is used for carrying UL data and NPUSCH F2 is used for transmitting acknowledgements of DL data. UL works either with a 3.75 or 15 kHz subcarrier spacing which is decided by the eNB. As the symbol duration for 3.75 kHz subcarrier spacing is four times higher compared to the 15 kHz spacing, it results in slot length of 2 ms. The slot length of 15 kHz is 0.5 ms. For NPUSCH F1 and 15 kHz subcarrier spacing, the resource unit (RU) duration is up to 8 ms that depends on the number of slots. Whereas, for 3.75 kHz, RU consists of one subcarrier in the frequency range, and 16 slots in the time range, therefore it has a duration of 32 ms. For NPUSCH F2, the RU has one subcarrier with 4 slots. Consequently, for the 15 kHz subcarrier spacing the RU has a 2 ms duration and for the 3.75 kHz subcarrier spacing 8 ms. Depending on the coverage level, the eNB indicates the NPUSCH repetition count that is maximum up to 128.

The allowed values of some parameters needed to configure the UE to successfully receive or transmit data are defined in Table I.

IV. PSM and eDRX NB-IoT Model

In this section, we propose the analytical system model to analyze energy consumption and latency. We consider that an eNB serves multiple UEs in an NB-IoT system. For ease of exposition, we make the following assumptions.

- The eNB and UEs have a finite buffer to store UL and DL traffic.
- The random access contention is not modeled, assuming an average value instead, as its effect on latency and energy consumption is negligible compared to that of PSM and eDRX.
- The three main states of a UE are considered to be PSM, eDRX, and RRC connected.
- The data packets arrive at the eNB or UE according to a Poisson process.
- The transmission time (UL or DL) for a packet is considered for fixed packet size, MCS, number of tones, tone spacing, number of slots and repetition count. However, the model can be configured with variable values of these parameters and the outputs can be used to determine the average power consumption and DL latency of the device.

Some important notations and parameters used in our model are listed in Table II. Based on the assumptions above, we consider a system to start from any of the states.

A. Introduction of state changes

The considered UE states are PSM, eDRX, and RRC Connected. The system behaves as follows based on the UE state.

- *PSM state*: This state starts with empty buffers on both sides. The maximal duration of the PSM state is fixed and denoted by T_{PSM}. This is the time a UE is in a deep sleep and equal to $T_{3412} - T_{3324}$. During the PSM state, as long as no UL packets are generated, DL packets generated are stored at the eNB buffer. When a UL packet is generated, the system switches to the RRC connected state. On the expiration of the PSM timer, if there exist

TABLE I: NB-IoT settings

Parameter name	Symbol	Possible Settings
Scheduled bandwidth	BW	{3.75, 15, 45, 90, 180} kHz
RU length of NPUSCH-F2	T_{RU}^{ack}	2 ms (if BW=15 kHz); 8 ms (if BW=3.75 kHz)
RU length of NPUSCH-F1	T_{RU}	8 ms (if BW=15 kHz); 32 ms (if BW=3.75 kHz)
Number of RU	N_{RU}	{1, 2, 3, 4, 5, 6, 8, 10}
Number of NPDSCH SFs	N_{SF}	{1, 2, 3, 4, 5, 6, 8, 10}
MCS index	MCS	{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
Maximum repetition of NPDCCH	R_{max}	2048
N/Ack repetition count (NPUSCH-F2)	N_{Rep}^{ack}	{1, 2, 4, 8, 16, 32, 64, 128}
Msg3 PUSCH repetition count	N_{Rep}^{Msg3}	{4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048}
NPUSCH repetition count	N_{Rep}^{NPUSCH}	{1, 2, 4, 8, 16, 32, 64, or 128}
NPDSCH repetition count	N_{Rep}^{NPDSCH}	{1, 2, 4, 8, 16, 32, 64, 128, 192, 256, 384, 512,
		768, 1024, 1536, 2048}
NPDCCH repetition count	N_{Rep}^{NPDCCH}	{1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048}
Preamble repetition count	N_{Rep}^{pmbl}	{1, 2, 4, 8, 16, 32, 64, 128}
RACH preamble latency	T_{pmbl}	5.6 ms (format 0); 6.4 ms (format 1)
Subframes occupied by NPRACH	N_{NPRACH}	$T_{pmbl} \cdot N_{Rep}^{pmbl}$
NPRACH resource periodicity	$P_{NPRACH} = N_{frame}^{UL}$	{40, 80, 160, 240, 320, 640, 1280, 2560} ms
NPUSCH-F1 scheduling delay	k_{N0}	{8, 16, 32, 64} ms
NPDSCH scheduling delay	k_{N1}	{0, 4, 8, 12, 16, 32, 64, 128} ms (if R_{max} <128)
		{0, 16, 32, 64, 128, 256, 512, 1024} ms (if R_{max} ≥128)
Scheduling delay between Rx data and Tx N/ACK	k_{rx_tx}	{12, 14, 16, 17} ms (if BW=15 kHz)
		{12, 20} ms (if BW=3.75kHz)

TABLE II: Notations

Symbol	Parameter name
λ_{UL}	Poisson arrival rate of UL packet
λ_{DL}	Poisson arrival rate of DL packet
$\lambda_{DL} + \lambda_{UL}$	Total poisson arrival rate, λ_{tot}
E_{UL}	Energy per unit time to transmit UL packet
E_{DL}	Energy per unit time to transmit DL packet
E_P	Energy needed to perform a paging action
E_C	Energy per unit time in Connected state
E_{PSM}	Energy per unit time in PSM state
E_{eDRX}	Energy per unit time in eDRX state
T_{RRC}	RRC inactivity Timer
$T_{eDRX} = T_{3324}$	eDRX state/Active Timer
T_{cycle}	eDRX cycle Timer
$T_{PSM} = T_{3412} - T_{3324}$	Deep sleep Time (PSM)
$\eta = T_{eDRX}/T_{cycle}$	Number of eDRX cycles
$\lambda_{tot} = \lambda_{UL} + \lambda_{DL}$	Total packet generation rate
N_{DL}	eNB buffer size
N_{UL}	UE buffer size
T_{DL}	Downlink Packet transmission time
T_{UL}	Uplink Packet transmission time

one or more DL packets in the DL buffer, the system switches to the RRC connected state, otherwise to the eDRX state.

- *RRC Connected state*: It is assumed that the generated packets at the eNB or the UE are transmitted immediately. There are two possibilities at the start of RRC Connected state, either the UL buffer is empty with some ready to transfer k_{DL} DL packets, where $1 \leq k_{DL} \leq N_{DL}$ or one UL packet and k_{DL} DL packets, where $0 \leq k_{DL} \leq N_{DL}$.
- *eDRX state*: Similar to the PSM state, the eDRX state starts with empty buffers. When T_{eDRX} expires, the system switches to the PSM state. However, if a UL packet is generated, the system switches immediately to the RRC connected state. The UE periodically performs paging to detect DL packets. The number of consecutive eDRX cycles is limited to a fixed value denoted by η (defined in Table II). If no UL packet is generated during an eDRX cycle and DL packets have been detected at paging, the system switches to the RRC connected state, and the packets are transmitted. Otherwise, the next eDRX cycle starts.

The energy consumption efficiency and its trade-off with latency can be derived after calculating the transition probability and the time duration of the different states, that together form a Markov chain.

B. RRC Connected state

The highest modulation of NB-IoT is Quadrature Phase Shift Keying (QPSK). In LTE, with the improvement in SINR, the nodes are capable of using higher modulation schemes, resulting in improved spectrum efficiency. However, in NB-IoT with improved SINR, the repetition factor reduces [23]. The transmission schemes and repetition counts are different for different channels of DL and UL. Therefore, we model both the packet transmissions separately.

1) Downlink: As shown in Figure 2, at every 20 SFs, only 15 SFs, can be used for NPDSCH or NPDCCH due to the presence of NPBCH, NPSS and NSSS. Therefore, based on the data transmitting SF position, it might need to wait up to 3 ms before transmitting. When the transmission collides with NSSS, it can take 3 ms due to waiting for NSSS and NPBCH to complete. Whereas, if it collides with NPBCH or NPSS, the transmission can be completed in 2 ms (during the next available SF). For other 15 SFs, the transmission can happen in that SF spending 1 ms. Therefore, the effective average data transmission time of 1 SF in the presence of NPSS, NSSS and NPBCH is given by Equation 1.

$$T_{DL}^{SF} = (1\ ms) \cdot \frac{N_{frame}^{DL} - (N_{NPBCH} + N_{NPSS} + N_{NSSS})}{N_{frame}^{DL}} + (2\ ms) \cdot \frac{N_{NPBCH} + N_{NPSS}}{N_{frame}^{DL}} + (3\ ms) \cdot \frac{N_{NSSS}}{N_{frame}^{DL}}\,, \tag{1}$$

where, N_{frame}^{DL} is the total DL frame size. N_{NPSS}, N_{NSSS} and N_{NPBCH} are the number of NPSS, NSSS and NPBCH packets, respectively, during the N_{frame}^{DL}. Therefore, considering N_{frame}^{DL} of 20 ms, the value of T_{DL}^{SF} is calculated as Equation 2.

$$\begin{aligned} T_{DL}^{SF} &= 1 \cdot \frac{20 - (2+2+1)}{20} \ + \ 2 \cdot \frac{2+2}{20} \ + \ 3 \cdot \frac{1}{20}\,, \\ &= \frac{26}{20}\ (ms) = 1.3\ (ms)\,. \end{aligned} \tag{2}$$

The application data is appended with to protocols headers. Generally, PDCP performs robust header compression to reduce the header size. The reduction and the compressed size of headers depend on the traffic type as defined in [19]. Assuming N_{data}^{DL} as the total size of application data and N_{header}^{DL} as the size of the headers including UDP, IP, PDCP, RLC, MAC, etc and TBS_{NPDSCH} as the transport block size for the NPDSCH resulting from the selection of MCS and number of SFs (N_{SF}) as defined in Table 16.4.1.5.1-1 in [21], the number of packet segments can be calculated. The packets are segmented at the RLC layer. Therefore, the data segment does not include the header size of the MAC layer (H_{mac}), which is appended after the segmentation. Therefore, the number of packet segments (MAC layer packets) is given by Equation 3.

$$N_{seg}^{DL} = \left\lceil \frac{N_{data}^{DL} + N_{header}^{DL} - H_{mac}}{TBS_{NPDSCH}(MCS, N_{SF}) - H_{mac}} \right\rceil. \tag{3}$$

The header size for the MAC layer in NB-IoT is 2 bytes [22]. To avoid the blockage by the DL resources, some transmission gaps (T_{DL}^{G}) are introduced on continuous DL transmission for T_{rx}^{C} [4]. Therefore, the time taken for each data segment transmission on the NPDSCH is given by Equation 4.

$$T_{NPDSCH} = T_{DL}^{SF} \cdot N_{SF} \cdot N_{Rep}^{NPDSCH} \cdot (1 \ + \ \frac{T_{DL}^{G}}{T_{rx}^{C}})\,, \tag{4}$$

where, N_{Rep}^{NPDSCH} is the number of repetitions for NPDSCH selected depending on the coverage area. Hence, the total transmission time of DL data (T_{DL}) for all the segments is calculated by adding the time of each activity as shown in Figure 3. That is for receiving scheduling information, receiving DL data, sending the corresponding acknowledgement and all the activity scheduling delays. Assuming the static TBS, this is given by Equation 5.

$$\begin{aligned} T_{DL} = N_{seg}^{DL} \cdot (T_{DL}^{SF} \cdot 1 \cdot N_{Rep}^{NPDCCH} \ + k_{N1} \ + \\ T_{NPDSCH} + \ k_{rx\_tx} + \ T_{RU}^{ack} \cdot N_{RU} \cdot N_{Rep}^{ack}) \\ + \ (N_{seg}^{DL} - 1) \cdot k_{next}^{DL}\,, \end{aligned} \tag{5}$$

where, k_{N1} and k_{rx_tx} are the scheduling delay required to change the channels as represented in Figure 3. Now, the energy consumption for DL data (E_{DL}^{C}) in RRC connected state is calculated as defined in Equation 6.

$$\begin{aligned} E_{DL}^{C} = N_{seg}^{DL} \cdot (T_{DL}^{SF} \cdot 1 \cdot N_{Rep}^{NPDCCH} \cdot E_{DL} \\ + \ k_{N1} \cdot E_{C} \ + \ T_{NPDSCH} \cdot E_{DL} \ + \\ k_{rx\_tx} \cdot E_{C} \ + \ T_{RU}^{ack} \cdot N_{RU} \cdot N_{Rep}^{ack} \cdot E_{UL}) \ + \\ (N_{seg}^{DL} - 1) \cdot k_{next}^{sch} \cdot E_{C}\,. \end{aligned} \tag{6}$$

2) Uplink: : The UL data time in RRC connected state includes the time spent for different activities, including the Random access (RA) procedure to send scheduling requests, receiving a UL grant, sending UL data and receiving an acknowledgment, as shown in Figure 3. Assuming N_{data}^{UL} as the total size of the UL application data, the number of UL packet segments (N_{seg}^{UL}) can be calculated similar to Equation 3 by replacing $TBS_{NPDSCH}(MCS, N_{SF})$ with $TBS_{NPUSCHF1}(MCS, N_{RU})$ which is defined as Table 16.5.1.2-2 in [21]. However, to maintain resynchronization with the DL reference signals, a certain gap (T_{UL}^{G}) is needed at a continuous NPUSCH transmission for T_{tx}^{C}. Whereas, for NPRACH, a 40 ms gap is introduced at every 64 preambles [24]. The calculation of the average NPUSCH transmission time of each UL data segment is given by Equation 7.

$$\begin{aligned} T_{NPUSCH} = T_{RU} \cdot N_{RU} \cdot N_{Rep}^{NPUSCH} \cdot \\ \frac{N_{frame}^{UL}}{N_{frame}^{UL} - N_{NPRACH}} \cdot (1 \ + \ \frac{T_{UL}^{G}}{T_{tx}^{C}})\,, \end{aligned} \tag{7}$$

where, N_{frame}^{UL} and N_{NPRACH} are the total UL frame size and number of NPRACH frames during the considered total UL frame.

The RA procedure consists of messages that occupy only one SF in the DL and one RU in the UL. There exists a four-step message transaction (Msg1, Msg2, Msg3, and Msg4) between the UE and the eNB. The time required to send Msg1 that transmits the NPRACH preamble and its timing is given by Equation 8.

$$T_{msg1} = T_{pmbl} \cdot 1 \cdot N_{Rep}^{pmbl}\,. \tag{8}$$

Then the control and data channels are transferred in DL as Msg2. There average transmission time is given by Equation 9.

$$\begin{aligned} T_{msg2} = T_{DL}^{SF} \cdot 1 \cdot N_{Rep}^{NPDCCH} \ + \\ T_{DL}^{SF} \cdot 1 \cdot N_{Rep}^{NPDSCH}\,. \end{aligned} \tag{9}$$

Msg3, where the RRC connection request and the acknowledgement of Msg2 are sent as UL transmission. Their transmission times (T_{msg3}) and (T_{UL}^{ack}) can be calculated using Equation 7 for their repetition count N_{Rep}^{Msg3} and N_{Rep}^{ack}, respectively. Also, Msg4 is similar to Msg2 that is an NPDCCH

TABLE III: State transitions

From ($State$, k_{UL}, k_{DL}, tl_{eDRX})	**To** ($State$, k_{UL}, k_{DL}, tl_{eDRX})	**Probability**
$(PSM,0,0,0)$	$(RRC,0,k_{DL},0)\,,0<k_{DL}<N_{DL}$	$Pn\left(\lambda_{DL},k_{DL},T_{PSM}\right)\cdot Pn\left(\lambda_{UL},0,T_{PSM}\right)$
$(PSM,0,0,0)$	$(RRC,0,N_{DL},0)$	$RPn\left(\lambda_{DL},N_{DL},T_{PSM}\right)\cdot Pn\left(\lambda_{UL},0,T_{PSM}\right)$
$(PSM,0,0,0)$	$(RRC,1,k_{DL},0)\,,0\leq k_{DL}<N_{DL}$	$A\left(k_{DL},1,T_{PSM}\right)$
$(PSM,0,0,0)$	$(RRC,1,N_{DL},0)$	1-$Pn\left(\lambda_{UL},0,T_{PSM}\right)-\sum_{k_{DL}=0}^{N_{DL}-1}A\left(k_{DL},1,T_{PSM}\right)$
$(PSM,0,0,0)$	$(eDRX,0,0,1)$	$Pn\left(\lambda_{tot},0,T_{PSM}\right)$
$(RRC,0,k_{DL},0)\,,0<k_{DL}\leq N_{DL}$	$(eDRX,0,0,1)$	$Pn\left(\lambda_{tot},0,T_{RRC}\right)$
$(RRC,0,k_{DL},0)\,,0<k_{DL}\leq N_{DL}$	$(RRC,1,0,0)$	$(\lambda_{UL}/\lambda_{tot})\cdot\left(1-Pn\left(\lambda_{tot},0,T_{RRC}\right)\right)$
$(RRC,0,k_{DL},0)\,,0<k_{DL}\leq N_{DL}$	$(RRC,0,1,0)$	$(\lambda_{DL}/\lambda_{tot})\cdot\left(1-Pn\left(\lambda_{tot},0,T_{RRC}\right)\right)$
$(RRC,1,k_{DL},0)\,,0\leq k_{DL}\leq N_{DL}$	$(eDRX,0,0,1)$	$Pn\left(\lambda_{tot},0,T_{RRC}\right)$
$(RRC,1,k_{DL},0)\,,0\leq k_{DL}\leq N_{DL}$	$(RRC,1,0,0)$	$(\lambda_{UL}/\lambda_{tot})\cdot\left(1-Pn\left(\lambda_{tot},0,T_{RRC}\right)\right)$
$(RRC,1,k_{DL},0)\,,0\leq k_{DL}\leq N_{DL}$	$(RRC,0,1,0)$	$(\lambda_{DL}/\lambda_{tot})\cdot\left(1-Pn\left(\lambda_{tot},0,T_{RRC}\right)\right)$
$(eDRX,0,0,tl_{eDRX})\,,1\leq tl_{eDRX}\leq\eta$	$(eDRX,0,0,tl_{eDRX}+1)$	$Pn\left(\lambda_{tot},0,T_{cycle}\right)$
$(eDRX,0,0,\eta)$	$(PSM,0,0,0)$	$Pn\left(\lambda_{tot},0,T_{cycle}\right)$
$(eDRX,0,0,tl_{eDRX})\,,1\leq tl_{eDRX}\leq\eta$	$(RRC,0,k_{DL},0)\,,0<k_{DL}<N_{DL}$	$Pn\left(\lambda_{DL},k_{DL},T_{cycle}\right)\cdot Pn\left(\lambda_{UL},0,T_{cycle}\right)$
$(eDRX,0,0,tl_{eDRX})\,,1\leq tl_{eDRX}\leq\eta$	$(RRC,0,N_{DL},0)$	$RPn\left(\lambda_{DL},N_{DL},T_{cycle}\right)\cdot Pn\left(\lambda_{UL},0,T_{cycle}\right)$
$(eDRX,0,0,tl_{eDRX})\,,1\leq tl_{eDRX}\leq\eta$	$(RRC,1,k_{DL},0)\,,0<k_{DL}<N_{DL}$	$A\ \left(k_{DL},1,T_{cycle}\right)$
$(eDRX,0,0,tl_{eDRX})\,,1\leq tl_{eDRX}\leq\eta$	$(RRC,1,N_{DL},0)$	1- $Pn\left(\lambda_{UL},0,T_{cycle}\right)-\sum_{k_{DL}=0}^{N_{DL}-1}A\left(k_{DL},1,T_{cycle}\right)$

followed by the NPDSCH, so its transmission time (T_{msg4}) is the same as that of Msg2.

The total time of the RA (T_{RA}) procedure can be calculated as the sum of all the four RACH messages and scheduling delays between them. This is given by Equation 10.

$$\begin{aligned}T_{RA}=T_{msg1}\ +\ k_{tx\_rx}\ +(k_{N1}\ +T_{msg2})\ +\\ k_{rx\_tx}\ +\ T_{msg3}\ +\ k_{tx\_rx}\ +\ (k_{N1}\ +\ T_{msg4})\\ +\ k_{rx\_tx}\ +\ T_{UL}^{ack}\,.\end{aligned}\tag{10}$$

Therefore, the energy of the RA procedure is given by Equation 11.

$$\begin{aligned}E_{RA}&=T_{msg1}\cdot E_{UL}+k_{tx\_rx}\cdot E_C+k_{N1}\cdot E_C\\&+T_{msg2}\cdot E_{DL}+k_{rx\_tx}\cdot E_C\\&+T_{msg3}\cdot E_{UL}+k_{tx\_rx}\cdot E_C+k_{N1}\cdot E_C\\&+T_{msg4}\cdot E_{DL}+k_{rx\_tx}\cdot E_C+T_{UL}^{ack}\cdot E_{UL}\,.\end{aligned}\tag{11}$$

The time to receive the UL grant or acknowledgement in the NPDCCH (T_{NPDCCH}) is calculated as given in Equation 12.

$$T_{NPDCCH}=T_{DL}^{SF}\cdot 1\cdot N_{Rep}^{NPDCCH}\,,\tag{12}$$

So, the total transmission time to transmit UL data is given by Equation 13.

$$\begin{aligned}T_{UL}=N_{seg}^{UL}\cdot(T_{RA}+k_{tx\_rx}+T_{NPDCCH}+k_{N0}\\+T_{NPUSCH}+k_{tx\_rx}+T_{NPDCCH})\ +\\(N_{seg}^{UL}-1)\cdot k_{next}^{sch}\,,\end{aligned}\tag{13}$$

where, k_{next}^{sch} are the scheduling delay for next UL grant transmission. Therefore, the energy to transmit UL data (E_{UL}^{C}) in RRC connected state is given by Equation 14.

$$\begin{aligned}E_{UL}^{C}=N_{seg}^{UL}\cdot(E_{RA}\ +\ k_{tx\_rx}\cdot E_C\ +\ T_{NPDCCH}\cdot E_{DL}\\+\ k_{N0}\cdot E_C\ +\ T_{NPUSCH}\cdot E_{UL}+k_{tx\_rx}\cdot E_C\ +\\T_{NPDCCH}\cdot E_{DL})\ +\ (N_{seg}^{UL}-1)\cdot k_{next}^{sch}\cdot E_C\,.\end{aligned}\tag{14}$$

C. eDRX and PSM state

This section discusses the model to calculate the latency of DL data traffic. We describe the Markov stochastic process by means of the following variables:

- *State*: It represents the UE states which encompass PSM, eDRX, or RRC connected.
- *DL buffer occupancy*: It represents the number of DL packets stored in the eNB transmission queue. New DL packets arriving at a full buffer are lost and do not contribute to the DL packet delay. This is represented by k_{DL}, such that, $0\leq k_{DL}\leq N_{DL}$.
- *UL buffer occupancy*: Similar to DL buffer occupancy, it represents the number of UL packets stored in the UE transmission queue and is denoted by k_{UL}, such that, $0\leq k_{UL}\leq N_{UL}$.
- *Counter eDRX intervals*: It represents the sequence number of the eDRX interval, denoted by tl_{eDRX}, such that, $0\leq tl_{eDRX}\leq\eta$. It is zero for the states PSM and RRC Connected.

The transitions between the states, with their corresponding probabilities, are given in Table III. The total number of states is given by $dim=2\times(1\ +\ N_{DL})\ +\ \eta$. The probability of N or more arrivals in an interval T and rate λ is given as Equation 15.

$$Pn\left(\lambda,k,T\right)=e^{-\lambda\cdot T}\cdot\left(\lambda\cdot T\right)^{k}/k!\,.\tag{15}$$

And the probability of k or more arrivals in an interval T is given as Equation 16.

$$RPn\left(\lambda,N,T\right)=1-\sum_{k=0}^{N-1}Pn\left(\lambda,k,T\right)\,.\tag{16}$$

The probability that a UL arrival occurs after k_{DL} DL arrivals, but before a time interval of length T expires is given by Equation 17.

$$A(k_{DL}, 1, T) = \int_0^T \lambda_{UL} \cdot e^{-\lambda_{UL} \cdot t} \cdot Pn\left(\lambda_{DL}, k_{DL}, t\right) dt = \frac{\lambda_{UL}}{\lambda_{DL}} \cdot \left(\frac{\lambda_{DL}}{\lambda_{tot}}\right)^{k_{DL}+1} \cdot RPn\left(\lambda_{tot}, k_{DL}+1, T\right) \,, \tag{17}$$

where, $Pn\left(\lambda, k, T\right)$ is the probability of k arrivals in an interval T and rate λ and $RPn\ \left(\lambda, k, T\right)$ is the probability of k arrivals do not happen in an interval T.

The holding time of different states which represents the average time the system remains in a state before making a transition to another state is calculated below.

- *Holding time of (PSM,0,0,0):* There are two possibilities either there are no UL arrivals during the PSM state, or there are. The probability of the first case is $e^{-\lambda_{UL} \cdot T_{PSM}}$ with the holding time of the total duration which is T_{PSM}. Therefore the probability of a UL arrival is $1 - e^{-\lambda_{UL} \cdot T_{PSM}}$ with the holding time as given by Equation 18.

$$HT_{3412} = \int_0^{T_{PSM}} t \cdot \lambda_{UL} \cdot e^{-\lambda_{UL} \cdot t} dt = \frac{1}{\lambda_{UL}} - \frac{e^{-\lambda_{UL} \cdot T_{PSM}}}{\lambda_{UL}} - T_{PSM} \cdot e^{-\lambda_{UL} \cdot T_{PSM}} \,. \tag{18}$$

Hence, the average PSM Holding time, denoted by $H_{(PSM,0,0,0)}$ is given by Equation 19.

$$H_{(PSM,0,0,0)} = e^{-\lambda_{UL} \cdot T_{PSM}} \cdot T_{PSM} \ + \left(1 - e^{-\lambda_{UL} \cdot T_{PSM}}\right) \cdot HT_{3412} \,. \tag{19}$$

- *Holding time of* $(RRC, k_{UL}, k_{DL}, 0)$*:* Assuming at the start of an RRC state interval, there are k_{DL} DL and k_{UL} UL packets present in the buffer such that, $0 \leq k_{DL} \leq N_{DL}$ and $k_{UL} \in \{0, 1\}$. Then each of these $k_{DL} \ + \ k_{UL}$ packets needs to be transmitted, as well as the new packets that are generated during the transmission time of these packets. Therefore, the time needed to empty the buffers is called the busy period starting with k_{DL} DL and k_{UL} UL packets. The UL and DL buffers are dimensioned in such a way that the probability of queue drops is negligible. We obtain the time needed to empty the buffers by applying the formula for the average length of a busy period of the M/G/1 queue starting with $k_{DL} \ + \ k_{UL}$ packets is given by Equation 20.

$$BP_{M/G/1}(k_{DL} \ + \ k_{UL}) = \frac{k_{DL} \ + \ k_{UL}}{(1 - \lambda_{tot} \cdot T_p)} \cdot T_p \,, \tag{20}$$

where, the average service time is given by $T_p = (\lambda_{UL} \cdot T_{UL} + \lambda_{DL} \cdot T_{DL})/\lambda_{tot})$.

At the end of this busy period, there are three possibilities.

  - *No UL nor a DL arrival occurs during RRC Connected* $[0, T_{RRC}]$: This happens with the probability $e^{-(\lambda_{UL} \ + \ \lambda_{DL}) \cdot T_{RRC}}$ and the holding time, in this case, is T_{RRC}.
  - *An UL arrival occurs first during* $[0, T_{RRC}]$: Its probability is given by Equation 21 and the holding time is given by Equation 22.

$$P1 = \int_0^{T_{RRC}} \lambda_{UL} \cdot e^{-\lambda_{UL} \cdot t} \cdot e^{-\lambda_{DL} \cdot t} dt = \frac{\lambda_{UL}}{\lambda_{tot}} (1 - e^{\lambda_{tot} \cdot T_{RRC}}) \,. \tag{21}$$

$$HT1 = \int_0^{T_{RRC}} t \cdot \lambda_{UL} \cdot e^{-\lambda_{UL} \cdot t} dt = \frac{1}{\lambda_{UL}} - \frac{e^{-\lambda_{UL} \cdot T_{RRC}}}{\lambda_{UL}} - T_{RRC} \cdot e^{-\lambda_{UL} \cdot T_{RRC}} \,. \tag{22}$$

  - *A DL arrival occurs first during* $[0, T_{RRC}]$: Its probability is given by Equation 23 and the holding time is given by Equation 24.

$$P2 = \int_0^{T_{RRC}} \lambda_{DL} \cdot e^{-\lambda_{UL} \cdot t} \cdot e^{-\lambda_{DL} \cdot t} dt = \frac{\lambda_{DL}}{\lambda_{tot}} (1 - e^{\lambda_{tot} \cdot T_{RRC}}) \,. \tag{23}$$

$$HT2 = \int_0^{T_{RRC}} t \cdot \lambda_{DL} \cdot e^{-\lambda_{DL} \cdot t} dt = \frac{1}{\lambda_{DL}} - \frac{e^{-\lambda_{DL} \cdot T_{RRC}}}{\lambda_{DL}} - T_{RRC} \cdot e^{-\lambda_{DL} \cdot T_{RRC}} \,. \tag{24}$$

The holding time of the state $(RRC, k_{UL}, k_{DL}, 0)$ is given by Equation 25

$$H_{(RRC, k_{UL}, k_{DL}, 0)} = e^{-(\lambda_{UL} + \lambda_{DL}) \cdot T_{PSM}} \cdot T_{RRC} + (P1 \cdot HT1) \ + \ (P2 \cdot HT2) \,. \tag{25}$$

- *Holding time of* $(eDRX, 0, 0, tl_{eDRX})$*:* The computation of the holding time of $(eDRX, 0, 0, tl_{eDRX})$ is similar to that of (PSM,0,0,0). Hence, this holding time is given by Equation 26.

$$H_{(eDRX, 0, 0, tl_{eDRX})} = e^{-\lambda_{UL} \cdot T_{cycle}} \cdot T_{cycle} \ + \left(1 - e^{-\lambda_{UL} \cdot T_{cycle}}\right) \cdot \left\{\frac{1}{\lambda_{UL}} - \frac{e^{-\lambda_{UL} \cdot T_{cycle}}}{\lambda_{UL}} - T_{cycle} \cdot e^{-\lambda_{UL} \cdot T_{cycle}}\right\} \,. \tag{26}$$

The results of the holding times can help in determining the probability that a random instant falls in an interval that starts with the state $(S, k_{UL}, k_{DL}, tl_{eDRX})$, denoted by

$\nu_{(S,k_{UL},k_{DL},tl_{eDRX})}$. Let the parameter D be the average time between state transition instants as defined by Equation 27.

$$
\begin{aligned}
D = & H_{(PSM,0,0,0)} \cdot \pi_{(PSM,0,0,0)} \\
& + \sum_{n=1}^{N_{DL}} H_{(RRC,0,n,0)} \cdot \pi_{(RRC,0,n,0)} \\
& + \sum_{n=1}^{N_{DL}} H_{(RRC,1,n,0)} \cdot \pi_{(RRC,1,n,0)} \\
& + \sum_{n=1}^{\eta} H_{(eDRX,0,0,n)} \cdot \pi_{(eDRX,0,0,n)} \,,
\end{aligned} \tag{27}
$$

where, π is the steady-state vector of the embedded Markov chain S calculated as the left eigenvector corresponding to the eigenvalue 1. The holding times are defined by Equation 28-31 and the probabilities that a random instant falls in one of the states PSM, RRC or eDRX by Equation 32-34.

$$
\nu_{(PSM,0,0,0)} = H_{(PSM,0,0,0)} \cdot \pi_{(PSM,0,0)}/D \,. \tag{28}
$$

$$
\begin{gathered}
\nu_{(RRC,0,k_{DL},0)} = H_{(RRC,0,k_{DL},0)} \cdot \pi_{(RRC,0,k_{DL},0)}/D \,, \\
where, \ 1 \leq k_{DL} \leq N_{DL} \,.
\end{gathered} \tag{29}
$$

$$
\begin{gathered}
\nu_{(RRC,1,k_{DL},0)} = H_{(RRC,1,k_{DL},0)} \cdot \pi_{(RRC,1,k_{DL},0)}/D \,, \\
where, \ 0 \leq k_{DL} \leq N_{DL} \,.
\end{gathered} \tag{30}
$$

$$
\begin{gathered}
\nu_{(eDRX,0,tl_{eDRX})} = H_{(eDRX,tl_{eDRX})} \cdot \pi_{(eDRX,0,0,tl_{eDRX})}/D \,, \\
where, \ 1 \leq tl_{eDRX} \leq \eta \,.
\end{gathered} \tag{31}
$$

$$
P_{PSM} = \nu_{(PSM,0,0,0)} \,. \tag{32}
$$

$$
P_{RRC} = \sum_{k_{DL}=1}^{N_{DL}} \nu_{(RRC,0,k_{DL},0)} + \sum_{k_{DL}=0}^{N_{DL}} \nu_{(RRC,1,k_{DL},0)} \,. \tag{33}
$$

$$
P_{eDRX} = \sum_{tl_{eDRX}=1}^{\eta} \nu_{(eDRX,0,0,tl_{eDRX})} \,. \tag{34}
$$

D. Downlink data delay analysis

As the arrival process of DL packets is assumed to be Poisson distributed, the Poisson Arrivals See Time Averages (PASTA) property can be applied to compute the delay of an arriving DL packet when it arrives at a random time instant in any of the states.

1) In the PSM state: The waiting time of a packet consists of the remaining time of the PSM cycle and the transmission time of all DL or UL packets that were present in the queue upon its arrival. There are two possible scenarios in the PSM state.

- *No UL arrival occurs during the PSM interval:* As there is no UL, therefore only DL transmissions can happen. If the arrival instant time of a packet is t, which is relative to the start of the PSM state, then the remaining time of the PSM cycle is given by $T_{PSM} - t$. Also, if k packets are waiting in the queue, then an additional waiting time of $k \cdot T_{DL}$ needs to be considered. The probability that this occurs is given by Equation 35.

$$
P_{wDL} = \frac{1}{T_{PSM}} \cdot \frac{(\lambda_{DL} \cdot t)^k}{k!} \cdot e^{-\lambda_{DL} \cdot t} \,. \tag{35}
$$

Hence, assuming an infinite capacity buffer, the average waiting time is given by Equation 36.

$$
\begin{aligned}
WT_{3412} &= \sum_{k=0}^{\infty} \int_0^{T_{PSM}} (T_{PSM} - t) + k \cdot T_{DL} \cdot P_{wDL} \\
&= (T_{PSM} \ + \ T_{DL} \cdot \lambda_{DL} \cdot T_{PSM})/2 \,.
\end{aligned} \tag{36}
$$

The two components of the average waiting time include the average remaining time of the PSM cycle ($T_{PSM}/2$) and the transmission time of the packets that have arrived during this time. Therefore, the PSM delay component is then given by Equation 37.

$$
Delay^1_{PSM} = WT_{3412} \ + \ T_{DL} \,. \tag{37}
$$

And the probability that there is no UL arrival during the PSM time is given by Equation 38.

$$
P_{NoUL} = e^{-\lambda_{UL} \cdot T_{PSM}} \,. \tag{38}
$$

- *A UL arrival occurs during the PSM interval:* The probability that there is a UL arrival during the PSM time is $1 - P_{NoUL}$. To calculate the latency of DL packets, assume that the DL packet arrives at a random time instant between the start of the PSM cycle and a UL arrival. When the UL data is ready to be sent, the UE exits the PSM state. The average latency can be calculated using the time of a renewal process with the first moment or mean μ and variance σ^2 which is $X = \mu^2 \ + \ \sigma^2/2\mu$ where μ is equal to HT_{3412} and σ^2 is given by Equation 39.

$$
\begin{aligned}
\sigma^2 &= \int_0^{T_{PSM}} t^2 \cdot \lambda_{UL} \cdot e^{-\lambda_{UL} \cdot t} dt \,, \\
&= \frac{2}{(\lambda_{UL})^2} \cdot (1 - e^{-\lambda_{UL} \cdot T_{PSM}}) \\
&\quad - T^2_{PSM} \cdot e^{-\lambda_{UL} \cdot T_{PSM}} \\
&\quad - \frac{2}{\lambda_{UL}} \cdot T_{PSM} \cdot e^{-\lambda_{UL} \cdot T_{PSM}} \,.
\end{aligned} \tag{39}
$$

Therefore the average delay when there is a UL arrival during PSM is given by Equation 40.

$$
Delay^2_{PSM} = X \ + \ T_{DL} \cdot (\lambda_{DL} X \ + \ 1) \,. \tag{40}
$$

The average delay a DL packet that arrives during a PSM interval is then given by Equation 41.

$$
\begin{aligned}
D_{(PSM,0,0,0)} = & P_{NoUL} \cdot Delay^1_{PSM} \ + \\
& (1 - P_{NoUL}) \cdot Delay^2_{PSM} \,.
\end{aligned} \tag{41}
$$

2) In the eDRX state: Similar to the PSM scenario, the eDRX state delay is given by Equation 42.

$$\begin{aligned} D_{(eDRX0,0,tl_{eDRX})} = e^{-\lambda_{UL} \cdot T_{cycle}} \cdot [T_{cycle}/2 + \\ T_{DL} \cdot (\lambda_{DL} \cdot T_{cycle}/2 + 1)] + \\ (1 - e^{-\lambda_{UL} \cdot T_{cycle}}) \cdot Y + \\ T_{DL} \cdot (\lambda_{DL} Y + 1)\,, \end{aligned} \tag{42}$$

where, Y is the mean time interval between the DL arrival at a random instant and the UL arrival that ends the eDRX interval. Y is computed in the same way as X.

3) In the RRC connected state: There are two possibilities. Firstly, the packet arrives during the busy period when the $k_{UL} + k_{DL}$ packets are being transmitted, as well as those generated during these transmissions, or it arrives after that busy period, but before the RRC timer of length T_{RRC} expires. Using the result for the length of a busy period in the M/G/1 queue starting with $k_{DL} + k_{UL}$ packets, the probability that the packet arrives during the busy period is given by Equation 43.

$$P_{DuringBP} = \frac{BP_{M/G/1}(k_{DL} + k_{UL})}{BP_{M/G/1}(k_{DL} + k_{UL}) + HT2}\,. \tag{43}$$

And the probability that the packet arrives after the busy period is given by Equation 44.

$$P_{AfterBP} = \frac{HT2}{BP_{M/G/1}(k_{DL} + k_{UL}) + HT2}\,. \tag{44}$$

The mean residual lifetime of a renewal process with the first moment μ and variance σ^2 is given by $(\mu^2 + \sigma^2)/2\mu$ where moment and variance are defined as in Equation 45.

$$\begin{aligned} \mu &= (k_{DL} + k_{UL}) \cdot T_p/(1 - \lambda_{tot} T_p)\,. \\ \sigma^2 &= (k_{DL} + k_{UL}) \cdot \lambda_{tot} \cdot T_p^3/(1 - \lambda_{tot} T_p)^3\,. \end{aligned} \tag{45}$$

Hence, when a packet arrives during the busy period that started with $k_{DL} + k_{UL}$ packets, the residual time of the busy period is given by Equation 46.

$$\begin{aligned} RBP(k_{DL} + k_{UL}) &= \mu^2 + \sigma^2/2\mu \\ &= \frac{(k_{DL} + k_{UL}) \cdot (1 - \lambda_{tot} T_p) + \lambda_{tot} \cdot T_p}{2(1 - \lambda_{tot} T_p)^2} \cdot T_p\,. \end{aligned} \tag{46}$$

And the RRC delay component of an RRC interval that starts with k_{DL} packets in the DL buffer, $1 \leq k_{DL} \leq N_{DL}$, is given by Equation 47.

$$\begin{aligned} D_{RRC}(k_{DL} + k_{UL}) &= P_{DuringBP} \cdot \big(RBP\,(k_{DL} + k_{UL}) \\ &\quad + T_p\big) + P_{AfterBP} \cdot T_p \\ &= P_{DuringBP} \cdot RBP(k_{DL} + k_{UL}) \\ &\quad + T_p\,. \end{aligned} \tag{47}$$

As such, the average delay an arriving DL packet experiences is given by Equation 48.

$$\begin{aligned} Delay_{tot} = P_{PSM} \cdot D_{(PSM,0,0,0)} + \\ \sum_{k_{DL}=1}^{N_{DL}} \nu_{(RRC,0,k_{DL},0)} \cdot D_{RRC}(k_{DL} + 0) + \\ \sum_{k_{DL}=0}^{N_{DL}} \nu_{(RRC,1,k_{DL},0)} \cdot D_{RRC}(k_{DL} + 1) + \\ P_{eDRX} \cdot D_{(eDRX,0,0,tl_{eDRX})}\,. \end{aligned} \tag{48}$$

E. Energy consumption analysis

While in the PSM state, the system consumes $EC_{(PSM,0,0,0)} = E_{PSM}$ energy per time unit whereas in an eDRX interval, it consumes $EC_{(eDRX,0,0,tl_{eDRX})} = (E_{eDRX} + E_P/H_{(eDRX,0,0,tl_{eDRX})})$ per time unit. The energy consumption during the RRC connected state consists of the consumption in emptying the UL and DL buffers which are given by Equation 49.

$$\begin{aligned} E_{RRC1} = k_{UL} \cdot E_{UL}^C + k_{DL} \cdot E_{DL}^C + \\ BP_{M/G/1}(k_{DL} + k_{UL}) \cdot \lambda_{UL} \cdot E_{UL}^C + \\ BP_{M/G/1}(k_{DL} + k_{UL}) \cdot \lambda_{DL} \cdot E_{DL}^C\,. \end{aligned} \tag{49}$$

The total energy consumption per time unit, during an RRC interval that starts in $(RRC, k_{UL}, k_{DL}, 0)$ is given by Equation 50.

$$\begin{aligned} EC_{(RRC,k_{UL},k_{DL},0)} = \frac{E_{RRC1} + P1 \cdot E_{DL}^C + P2 \cdot E_{UL}^C}{H_{(RRC,k_{UL},k_{DL},0)}} \\ + E_C\,. \end{aligned} \tag{50}$$

Hence, the total energy consumption per time unit of the NB-IoT network is given by Equation 51.

$$\begin{aligned} E_{tot} = P_{PSM} \cdot EC_{(PSM,0,0,0)} + \\ \sum_{k_{DL}=1}^{N_{DL}} \nu_{(RRC,0,k_{DL},0)} \cdot EC_{(RRC,0,k_{DL},0)} + \\ \sum_{k_{DL}=0}^{N_{DL}} \nu_{(RRC,1,k_{DL},0)} \cdot EC_{(RRC,1,k_{DL},0)} + \\ P_{eDRX} \cdot EC_{(eDRX,0,0,tl_{eDRX})}\,. \end{aligned} \tag{51}$$

V. Numerical Results and Validation

This section presents the evaluation of the energy consumption and DL latency results of the analytical model and the Pareto optimized data points. Firstly, the description of the simulation setup is presented. This is followed by comparing the results of our analytical model and the simulation. Thereafter, these results are analyzed in detail based on the energy consumption and DL latency of a device. Finally, the optimized parameters of the power-saving schemes which are obtained using Pareto front analysis are discussed.

TABLE IV: Simulation and Model parameters

Parameters	Symbol	Value
Operating voltage	V_{op}	3.8
PSM state power consumption	E_{PSM}	$0.000003 \times V_{op} = 0.0000114\ W$
eDRX state power consumption	E_{eDRX}	$0.0008 \times V_{op} = 0.00304\ W$
Connected state power consumption	E_C	$0.006 \times V_{op} = 0.0228\ W$
Rx power consumption	E_{DL}^{C}	$0.023 \times V_{op} \times T_p = 0.0874 \times T_p\ J$
Tx power consumption	E_{UL}^{C}	$0.275 \times V_{op} \times T_p = 1.048 \times T_p\ J$
Paging state energy consumption per RB	E_P	$0.023 \times V_{op} \times T_{cch} = 0.0874\ \times T_{cch}\ J$
Tx time on control channel per RB	T_{cch}	$0.214285\ ms$
Tx time on data channel per RB	T_{dch}	$0.92857\ ms$
Modulation scheme	MCS	9
Application data size	N_{data}^{DL}	64 bytes
Number of DL/UL packet segments	N_{seg}^{DL} or N_{seg}^{UL}	7
Packet transmission time	T_p	$T_{dch} \times N_{seg}^{DL} = 6.499$ ms
RRC inactivity timer	T_{RRC}	[1,5,10,20,30,40,50,60] s
eDRX state/Active timer	T_{eDRX}	[0, 20.48, 40,96, 81.92, 163,84, 327.68, 655.36, 1310.72, 2621.44, 5242.88, 10485.76] s
eDRX cycle timer	T_{cycle}	[0, 20.48, 40,96, 81.92, 163,84, 327.68, 655.36, 1310.72, 2621.44, 5242.88, 10485.76] s
PSM timer	T_{PSM}	[0, 136.219, 272.438, 544.877, 1089.755, 2179.511, 4359.023, 8718.046, 17436.093, 34872.187] s
Number of eDRX cycles	η	[1,2,4,8,16,32]

A. Simulation setup

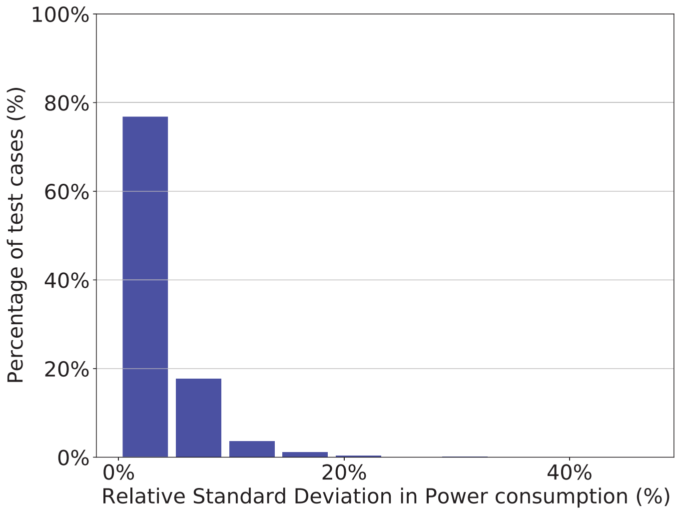
We use ns-3 to analyze the latency and power consumption of an NB-IoT UE using eDRX and PSM. The ns-3 simulator is one of the most popular computer network simulators, where some NB-IoT features have been implemented on top of the LTE code [25]. The code defines several new features, such as limiting the number of LTE physical resource blocks to one in the frequency domain, modifying the physical error model to adopt lower Modulation and Coding Schemes (MCS), separating the SFs for control and data channels, and including cross SF delays for both channels. We implemented the RRC idle mode features (PSM and eDRX), energy calculation module, and scripts to calculate latency. The complete description of the code implementation is described in our previous paper [17].

The simulation results under various parameter values are compared with the analytical model to evaluate its accuracy. The analytical model is solved using MATLAB. All of the simulation parameters are shown in Table IV, which are selected according to the NB-IoT specification. The reference power values are taken from the u-blox SARA-N3 NB-IoT radio module datasheet [29]. As mentioned in the datasheet, the current consumption in the PSM state of the SARA module is 3 μA at an operating voltage of 3.8 V. The power consumption of this state is calculated as 0.000003 $\times$ 3.8 that is 0.0114 mW. Similarly, the power consumption values of the other states are calculated and mentioned in Table IV. The transmission time on the control and data channel is evaluated from the ns-3 results. For simplicity, the experiments are performed for a single paging occasion using one device and one eNB. Many IoT use cases have data intervals ranging from weeks to months; therefore, multiple paging occasions in each eDRX cycle waste the energy in monitoring them. Furthermore, the repetition count only increases the duration of RRC connected state. Therefore, to focus the effect of the PSM and eDRX feature on the DL latency and power consumption, the repetition of control and data channels is ignored. However, it should be noted that the ns-3 mathematical model supports an arbitrary number of devices, paging occasions per eDRX cycle and repetitions.

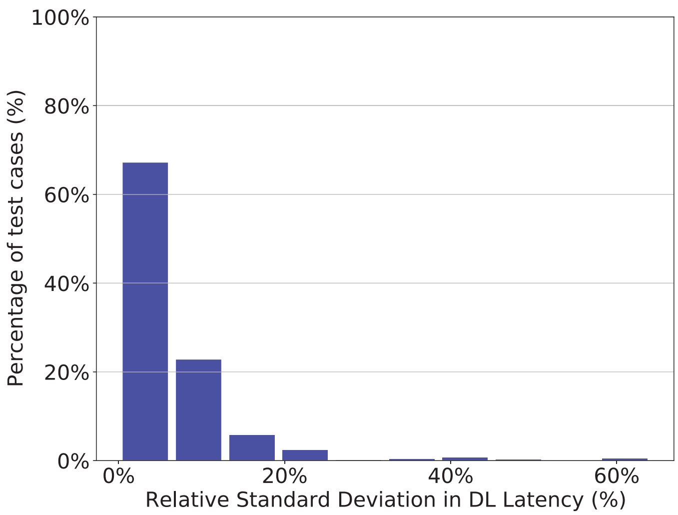
B. Model validation

We ran a total of 1826 test cases to compare the result of the analytical model and simulation. The test cases are created from the parameters mentioned in Table IV. These test cases analyze only DL data as well as a combination of UL and DL data. We have considered the data interval of 1 minute and 60 minutes. The histograms shown in Figure 4 present the relative standard deviation in power consumption and DL latency between the simulation and the model results. There is a relative standard deviation of 3.16% in power consumption and 6.37% in DL latency found in a run of all the test cases. The mean difference is around 4.45% and 8.73% in power consumption and DL latency respectively. The relative difference in power consumption value is high in the cases where there is a high probability of a UE to remain in PSM state. Around 0.49% of test cases have a deviation in power consumption value of more than 20%. But most of these cases have very small (less than 1 mW) absolute power consumption value. Therefore, a small value difference results in a large relative deviation.

The large relative deviation in latency value ($\geq$ 40%) was also observed for large values of the PSM timer with frequent DL data, but T_{eDRX} (Active timer) is 655.36 s. In 4.3% of test cases, there is a relative standard deviation above 20%. Mostly these cases have the eDRX cycle time of 40.96 s with large PSM and Active timer values, having DL data interval (DLI) or/and UL data interval (ULI) of 60 minutes. There are no test cases where the relative standard deviation for both (the latency and the power consumption) is higher than 15%.

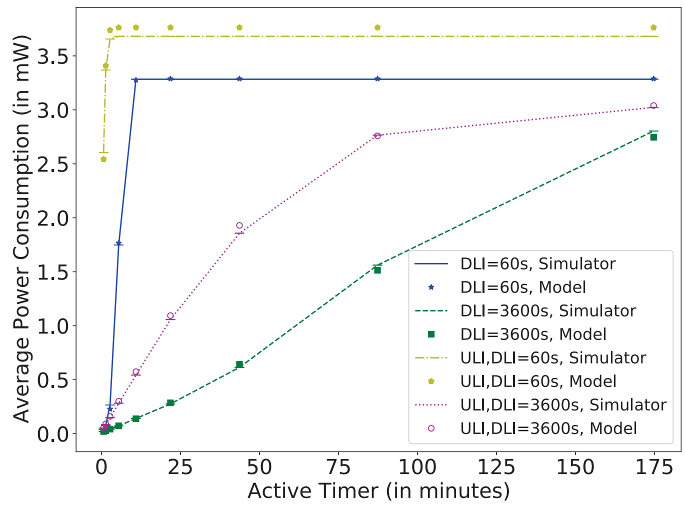


(a) Power consumption

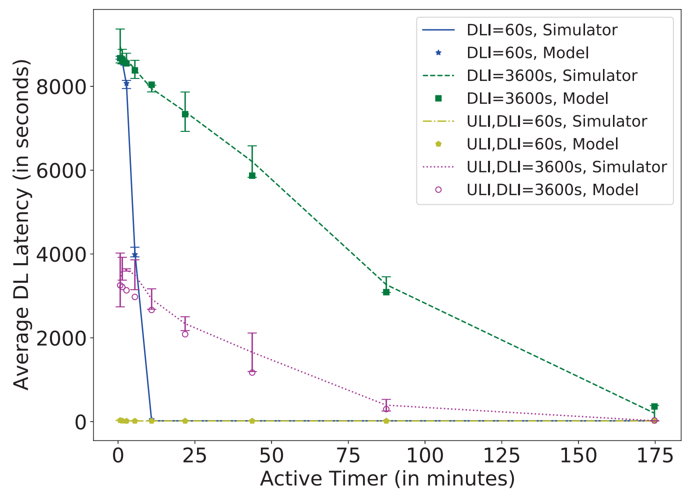


(b) DL Latency

Fig. 4: Histogram of percentage relative standard deviation between Model and Simulation results



(a) Power consumption



(b) DL Latency

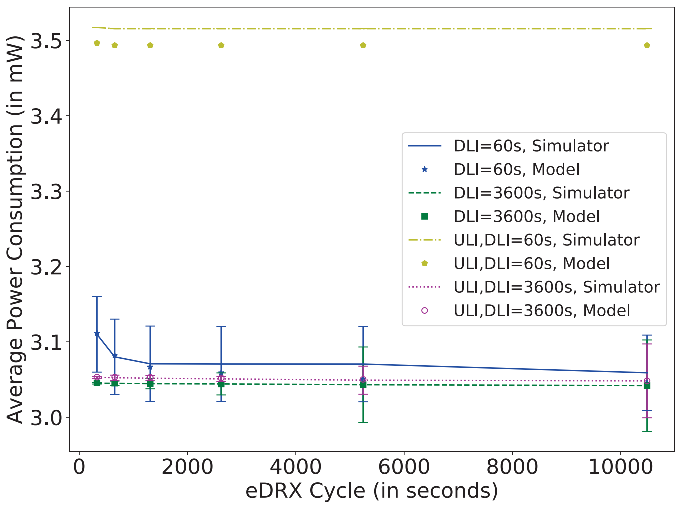
Fig. 5: Comparing Simulator and Model for different data intervals varying Active timer (T_{3324}) for parameters: RRC inactivity timer (T_{RRC}) = 1 s, eDRX cycle (T_{cycle}) = 40.96 s, and PSM timer (T_{3412}) = 4.84 hours

A clear observation is that the deviation in the results (for both energy consumption and DL latency) of the model and simulation is higher, mostly when the PSM timer is large in combination with certain other timer values. Also, the accuracy of the model is higher in terms of power consumption than in terms of DL latency.

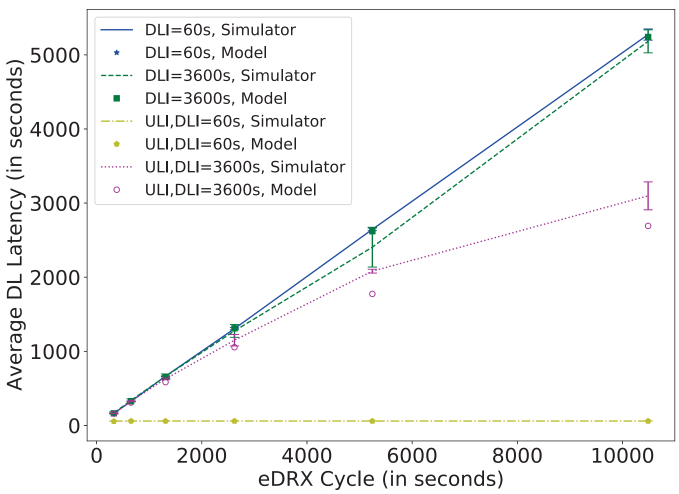
Moreover, different set of simulation results are also collected for five different seed values and ranges to generate various Poisson distributions in data transmission timings. The standard deviation observed in the simulation results are 0.18% and 4.72% for the power consumption and the DL latency respectively. These observed variation improves the relative standard deviation between the test results of simulation and the analytical model.

C. Result analysis

Figures 5 to 8 plot the results of the analytical model and the ns-3 simulation, for various transmission rates. It can be visibly seen that the analytical model and simulation follow the same behavior, and the results match closely in all the plotted graphs. It can be observed that as the DLI increases, the power consumption decreases since fewer packets are being transferred per unit time. Moreover, the addition of UL data transmissions to DL data transmissions further increases the power consumption and decreases the DL latency because the UE gets an early opportunity to transition to the RRC connected state. These figures denote the effect on power consumption and DL data latency by varying different timers. The effect of the Active timer and eDRX cycle timer is analyzed in Figure 5 and 6 respectively. These parameters are important when IoT use case expects infrequent DL data. Whereas, when DL data is sent frequently, the RRC inactivity timer influences the DL latency. Its effect is shown in Figure 7. Finally, the PSM timer plays an important role in saving maximum energy keeping the device in a deep sleep state. Figure 8 shows the variation of PSM timer and its effect on

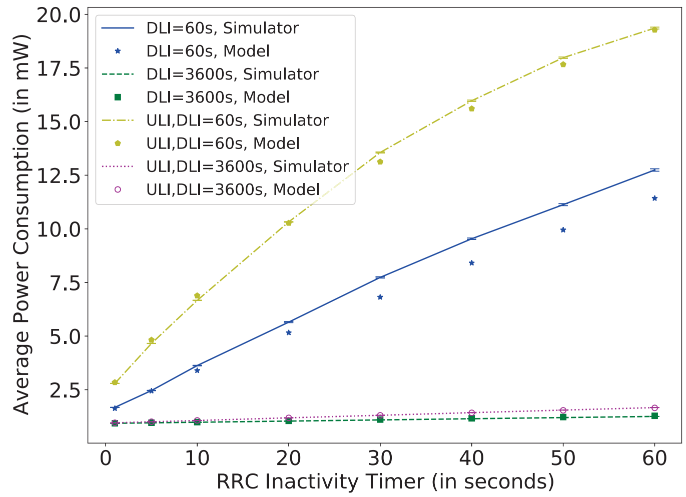


(a) Power consumption

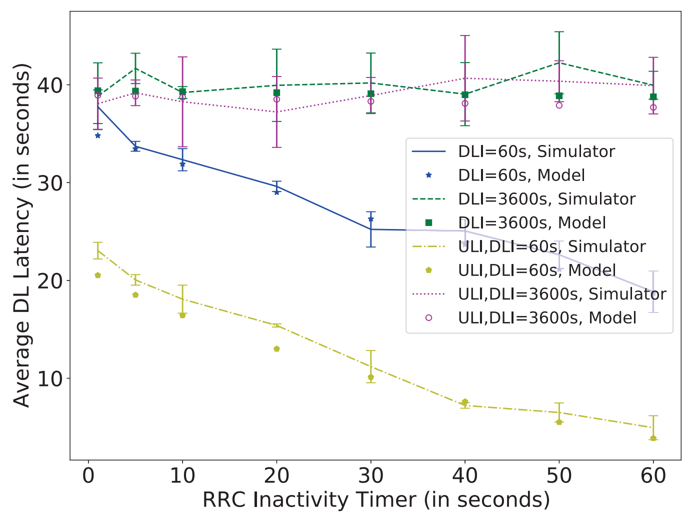


(b) DL Latency

Fig. 6: Comparing Simulator and Model for different data intervals varying eDRX cycle timer (T_{cycle}) for parameters: T_{3324} = 10485.76 s, T_{RRC} = 1 s, and PSM Disabled



(a) Power consumption



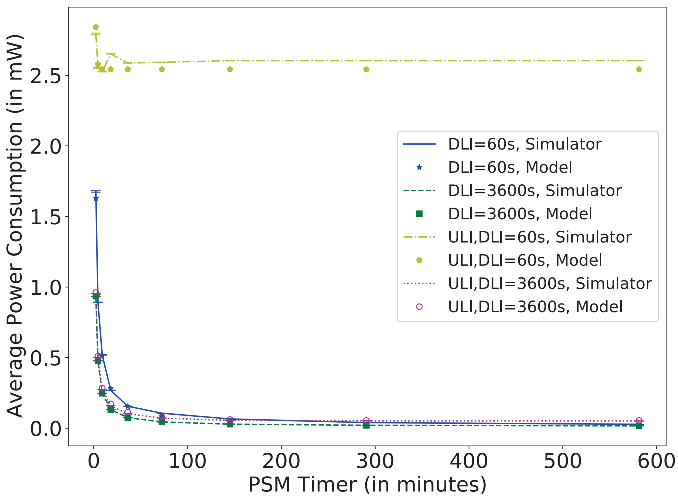
(b) DL Latency

Fig. 7: Comparing Simulator and Model for different data intervals varying RRC inactivity timer (T_{RRC}) for parameters: T_{3324} = T_{cycle} = 40.96 s, T_{3412} = 136.219 s

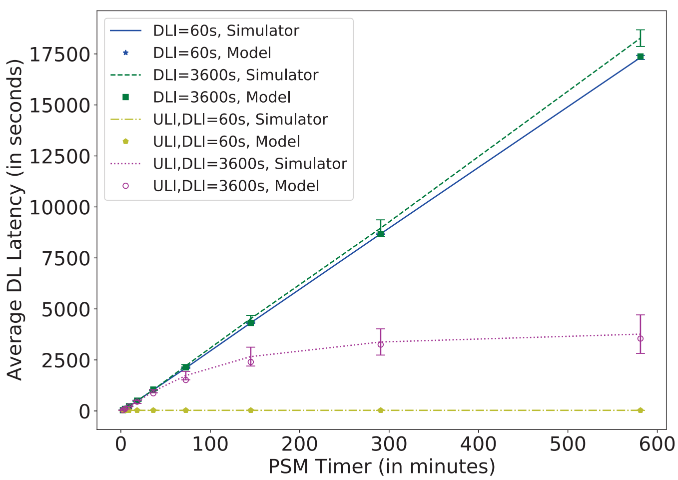
both power consumption and DL latency.

In Figure 5, the Active timer is varied from 20.48 seconds to 2.91 hours. Since, in the RRC Idle state, the UE enters into eDRX state before PSM state, the PSM timer (T_{3412}) needs to be configured to a higher value than the Active timer (T_{3324}). Therefore, the PSM timer is configured to 4.84 hours while other timers are configured with small values. We can observe from Figure 5a that the Active timer mostly impacts the power consumption linearly; except for frequent transmission intervals where large values of the Active timer have nearly no effect on power consumption and DL latency. This is because as the Active timer increases, the UE gets more time to remain in the eDRX state and do paging. During the paging (at every eDRX cycle time), the UE gets an opportunity to receive the incoming DL data keeping the DL latency low. It is observed from Figure 5b, the minimum latency is 20.24 s when only DL data is transferred (blue line) and 16.22 s for a combination of both DL and UL (yellow line). Also, the maximum latency is around 2.36 hours for a low value of the Active timer as in this case, the probability that the UE remains in the PSM state is high, which dominates the latency value (which is around half of the PSM timer). Therefore, use cases with infrequent UL traffic, and that require DL latencies in the order of only tens of seconds, should be configured with a long Active timer, and short PSM deep-sleep time.

The graph plotted in Figure 6 shows the variation in DL data latency and power consumption for different eDRX cycle timer values by fixing the Active timer and RRC inactivity timer. The PSM feature is disabled. A large value of the Active timer is chosen to increase the probability of the UE to be in eDRX state so that the effect of the connected state is minimized. The interesting observation is that the power consumption does not vary much, but the latency increases linearly by increasing the eDRX cycle when only DL data is sent. The reason is that the UE sleeps, consuming low energy until the next paging occasion, and this paging occasion time affects the



(a) Power consumption



(b) DL Latency

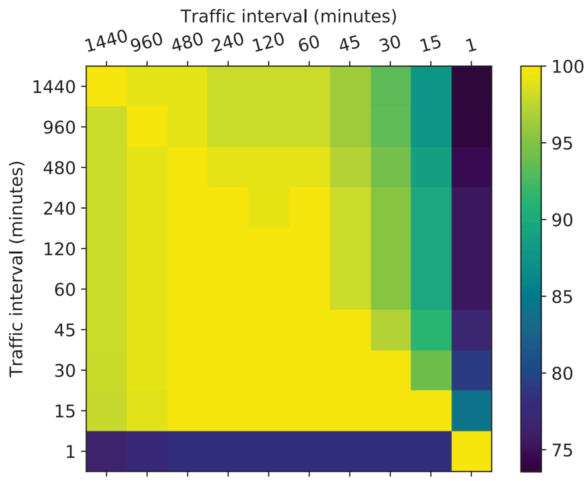
Fig. 8: Comparing Simulator and model for different data intervals varying PSM timer (T_{3412}) for parameters:
T_{RRC} = 1 s, T_{3324} = T_{cycle} = 40.96 s

DL data latency. A large value of the eDRX cycle timer means that the paging occurrence is delayed, thereby increasing the latency without affecting the power consumption substantially. The variations in power consumption are observed due to the transition times from RRC connected to Idle state. When UL data is sent with DL data, the UE switches to the connected mode for sending UL data without waiting for the paging occasion. Therefore, the eDRX cycle timer has nearly no effect on DL latency when the ULI is close to the eDRX cycle time (as seen for the yellow line where ULI = DLI = 60s). The DL latency is nearly half of the eDRX cycle value. Considering the above observations, we can conclude that for IoT use cases that require low latency, a low value of the eDRX cycle value should be suggested, especially if the UL transmission interval is high. In Figure 6a, the standard deviation in simulation result for DL interval of 60s is visibly seems large but the average deviation is only 0.47%.

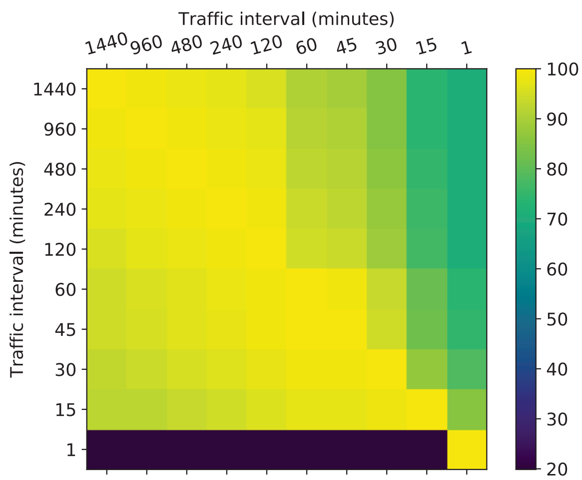
A variation of the RRC inactivity timer is shown in Figure 7 with a small fixed value of the Active timer, eDRX cycle time, and PSM timer. Figure 7a shows that as RRC inactivity timer increases, the power consumption increases because after each data transmission, the UE needs to stay in RRC connected state until the RRC inactivity timer expires. The RRC inactivity timer seems to have less effect on the power consumption for higher data interval scenarios because the UE spends more time in the RRC Idle state than in connected. It has an opposite effect on DL latency, i.e., it decreases with an increase in RRC inactivity timer (cf. Figure 7b). By analyzing the blue lines of Figure 7, it is observed that when DLI is 1 minute, the RRC inactivity timer affects the power consumption more than the DL latency. When the RRC inactivity timer increases from 1 second to 1 minute, it is observed that there is around 278% increase in power consumption and 50% decrease in DL latency. In Figure 7b, there is visible variation in the model and simulation results, it is not more than 10% that is when DLI and ULI are 60 s (yellow line) and sometimes this reduces when we compare with the results of the simulator executed with different seed values. It can be concluded that the RRC inactivity timer should be configured carefully by the network operator as it augments the power consumption with comparatively little improvement on DL latency. This effect is most noticeable when there are frequent UL transmissions.

Finally, we discuss the test cases where the UE is configured such that it remains in the PSM state as much as possible. Figure 8 shows the power consumption and DL latency variation as a function of the PSM timer with small values of Active timer, eDRX cycle timer and RRC inactivity timer. It can be seen that the power consumption decreases exponentially with an increase in PSM timer. Whereas, if there is only downlink it is linear following the PSM timer, while if there is UL transmission, the latency is bounded by the UL interval. This occurs because the scheduled UL data stops the RRC idle state, and therefore, the PSM timer has minimal effect on latency and power consumption. The eDRX cycle timer and PSM timer follow a similar trend in the variation of DL latency. The DL latency is nearly half the PSM timer in a DL only scenario. It also shows that a large value of the PSM timer (>4.84 hours) does not substantially affect the power consumption since most of the DL packets are waiting in a queue for the expiration of the PSM timer. However, as some of the packets are waiting, the DL latency increases with the PSM timer. Therefore, when the DLI is infrequent, the PSM timer should be configured based on the maximum required latency of an application.

Analyzing the graphs of different timers, it can be concluded that the IoT use cases with frequent DL data requirements should consider small eDRX cycle timer values. This is because, with the increase in this timer, the DL latency increases with nearly no effect on power consumption. However, we need to consider the trade-off between power consumption and latency when selecting Active timer, PSM timer, and RRC inactivity timer. If the IoT use case has frequent UL data, small values of eDRX cycle and RRC inactivity timer should be selected, and the optimal values of Active timer and PSM timer need to be calculated depending on the UL data frequency.



(a) DL only



(b) DL and UL

Fig. 9: HeatMap of Pareto front matching percentage for different traffic intervals

D. PSM and eDRX parameter optimization

The problem of obtaining optimal parameter values for different use cases is discussed here. We define a use case in terms of its UL and DL transmission rate. It involves two objective functions as DL latency, and power consumption needs to be minimized. This objective using multi-objective optimization is solved using Python by determining the Pareto front to evaluate the latency and power consumption metrics for different values of all the timer combinations. We plot an approximate Pareto front obtained by solving our proposed Markov Model for 6560 test cases for the RRC inactivity timer (T_{RRC}) ranging from 1 to 60 s, Active timer (T_{3324}) and eDRX cycle from 0 to 10485.76 s, and PSM timer (T_{3412}) from 0 to 35709120 s. All 6560 test cases were run for different DLI and ULI (1 minute to 24 hours).

Table V lists the number of Pareto optimal data points among the 6560 test cases for each transmission interval (DL only and UL-DL). It can be observed that the number of

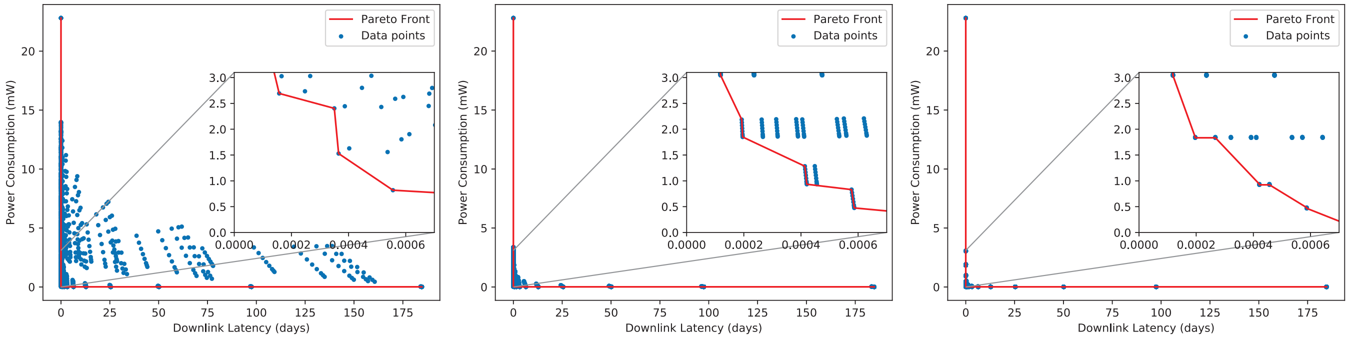
TABLE V: Number of Pareto front data points

Data Interval (#Minutes)	For DL only	For DL & UL
1440	106	96
960	107	96
480	107	96
240	106	96
120	106	96
60	106	92
45	104	91
30	101	87
15	95	77
1	102	332

Pareto front elements decreases as the data interval decreases except in the case of a 1 minute interval. Moreover, the fronts obtained at lower data intervals are mostly the subset of the previous test case (having a larger data interval), which can be observed from the heatmap shown in Figure 9. The heatmap indicates the number of matching Pareto optimal data points represented as colors ranging from yellow to blue. Yellow entities match heavily while blue ones overlap little or nothing. As such, in the DL only scenario, except the 1 and 60 minutes data intervals, all other intervals are a subset of its larger data intervals. However, in the DL and UL scenario, only 1 minute shows a disparity in the behavior.

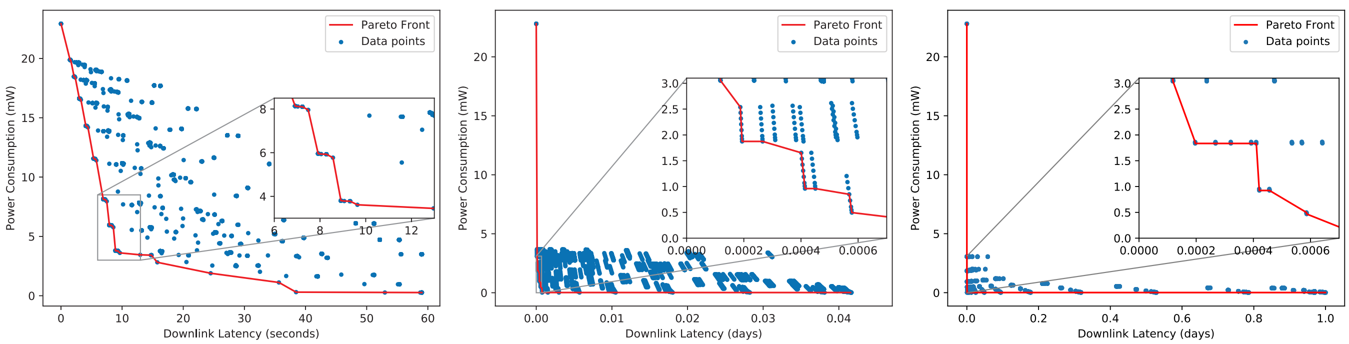
Figures 10 and 11 show the results in terms of power consumption and DL latency and plot the Pareto front for 6 traffic scenarios. The points on the red line are the Pareto front, whereas the bold blue dots represent the points for all the test cases. The goal is to search the points close to the bottom left (i.e., the elbow), which means minimal DL latency consuming minimal energy. The plots show that the minimal DL latency is achieved when PSM and eDRX are disabled spending high power (22.8 mW) because the UE is always in RRC connected state (top left point on each figure). The minimum power is consumed when only PSM is enabled by disabling the eDRX feature compromising latency optimization (bottom right). The pattern of the plots are similar for all cases (15 to 1440 minutes) but not when the data interval is 1 minute. The RRC inactivity timer impacts the power consumption more than DL latency, and this impact decreases with the increase in data frequency. Therefore, it can be visibly seen that the number of points present on the red line increases when the data interval increases from 1 to 60 minutes and later starts decreasing (which can be seen in Figure 10 or 11). It is due to the reduced effect of the RRC inactivity timer under more sporadic traffic (i.e., 1 hour and more) that the blue dots edge closer and closer to the red line of the Pareto front.

It is observed that most of the Pareto front data points obtained using a short eDRX cycle time of 20.48 seconds. We list some of the relevant optimal configuration values in Table VI. We can observe that the pattern of optimal configurations is similar, and we can point out that the bottom left cases have the smallest values of all the timers. For each type of traffic frequency, we marked a Pareto optimal configuration with a good latency-energy trade-off (i.e., near the bottom left of the graph) with a grey background. This grey colored configuration allows reducing the energy consumption to 4-



(a) DL 1 minute (b) DL 60 minutes (c) DL 1440 minutes

Fig. 10: Power consumption and DL latency for all 6560 test cases, with the Pareto front marked by a line, for DL only scenarios



(a) UL & DL 1 minute (b) UL & DL 60 minutes (c) UL & DL 1440 minutes

Fig. 11: Power consumption and DL latency for all 6560 test cases, with the Pareto front marked by a line, for DL and UL scenarios

10% of the max (i.e., from more than 22.8 down to less than 3 mW) while keeping the latency below 20 seconds. The change in data interval from 60 to 1440 minutes, does not affect the latency and power consumption much in comparison to changing it from 1 to 60 minutes. This change in data interval from 1 to 60 minutes influences the latency and power consumption with more than 55% when both DL and UL data is sent and more than 25% when only DL data is sent. Some optimal configurations have no effect on power consumption and latency when the Active timer range is between 20.48 and 655.36 s, provided the PSM feature is disabled. However, many configurations affect them by more than 50% as observed when the Active timer is reduced from 81.92 s to 40.96 s in case A, C or F. In the same way, the RRC inactivity timer affects them, but its effect decreases with an increase in data interval as noticed from case A and B. When the RRC inactivity timer is reduced from 60 to 20 seconds, the power consumption reduces and DL latency increases by more than 70% when DLI is 1 minute, but in case the data rate is increased to 60 minutes (case B) this effect is reduced to around 1%.

VI. Conclusion

In this paper, we defined an analytical model to calculate the power consumption and DL latency for NB-IoT with the PSM and eDRX power saving features. The model's accuracy was compared to ns-3 simulation results for various NB-IoT timers (i.e., Active timer, eDRX cycle timer, PSM timer, and RRC inactivity timer). The results derived from these two methods showed an average deviation of 6.33% in the results of power consumption and 8.73% in DL latency. The goal was to calculate the optimal configurations to obtain minimal power consumption and minimal DL latency. The graphs of different timers showed diverse behavior on power consumption and DL latency for data frequency and transmission direction. Therefore, to analyze the trade-off between both objectives, we used Pareto front analysis. This provides the optimal values of NB-IoT timers for LPWA use cases. It is found that most optimal configurations had small timer values for various data intervals. Moreover, the impact of the RRC inactivity timer on power consumption and DL latency for optimal configurations decreases with increasing data frequency. The LPWA applications where lower DL latency is preferable, a small value of the eDRX cycle and PSM timer should be considered. Whereas, for high UL frequency use cases, values of eDRX cycle and RRC inactivity timer should be small, as in these scenarios, increasing the timers has a little positive effect on DL latency, but does have a strong negative effect on energy consumption. Finally, if UL traffic is infrequent, the PSM timer should be set at most to double the maximum tolerable DL

TABLE VI: A subset of optimal Pareto front configurations

Test Case	RRC inact. (s)	Active timer(s)	PSM Timer (s)	eDRX cycle (s)	DL Latency (s)	Power cn. (mW)
	60	[20.48 - 655.36]	Disabled	20.48	4.59	13.96
	20	[20.48 - 655.36]	Disabled	20.48	7.89	7.58
	1	[20.48 - 655.36]	Disabled	20.48	10.1	3.32
A (DL 1 minute)	1	81.92	136.219	20.48	13.58	2.69
	1	40.96	136.219	20.48	31.54	1.52
	1	Disabled	136.219	Disabled	67.66	0.17
	1	Disabled	1089.755	Disabled	544.38	0.044
	60	[20.48 - 655.36]	Disabled	20.48	10.08	3.36
	20	[20.48 - 655.36]	Disabled	20.48	10.19	3.15
	1	[20.48 - 655.36]	Disabled	20.48	10.24	3.05
B (DL 60 minutes)	1	81.92	136.219	20.48	16.92	1.85
	1	20.48	136.219	20.48	50.66	0.48
	1	Disabled	136.219	Disabled	68.09	0.0178
	60	[20.48 - 655.36]	Disabled	20.48	10.239	3.054
	20	[20.48 - 655.36]	Disabled	20.48	10.244	3.045
	1	[20.48 - 655.36]	Disabled	20.48	10.246	3.041
C (DL 1440 minutes)	1	81.92	136.219	20.48	16.98	1.83
	1	40.96	136.219	20.48	36.38	0.92
	1	20.48	136.219	20.48	50.71	0.46
	60	655.36	[1089.755 - 35709120]	20.48	1.48	19.87
	60	[20.48 - 655.36]	Disabled	20.48	1.48	19.87
	20	655.36	[1089.755 - 35709120]	20.48	5.33	11.56
	20	[20.48 - 655.36]	Disabled	20.48	5.33	11.56
D (DL & UL 1 minute)	1	[20.48 - 655.36]	Disabled	20.48	8.91	3.80
	1	655.36	[1089.755 - 35709120]	20.48	8.91	3.80
	1	81.92	136.219	20.48	9.63	3.61
	1	20.48	136.219	20.48	24.48	1.89
	1	Disabled	1089.755	Disabled	59.01	0.26
	60	[20.48 - 655.36]	Disabled	20.48	9.89	3.68
	1	[20.48 - 655.36]	Disabled	20.48	10.22	3.05
	1	81.92	136.219	20.48	16.75	1.87
E (DL & UL 60 minutes)	1	20.48	136.219	20.48	49.93	0.49
	1	Disabled	136.219	Disabled	67.22	0.0179
	60	[20.48 - 655.36]	Disabled	20.48	10.23	3.06
	1	[20.48 - 655.36]	Disabled	20.48	10.24	3.04
	1	81.92	136.219	20.48	16.97	1.83
F (DL & UL 1440 minutes)	10	40.96	136.219	20.48	36.36	0.92
	1	Disabled	136.219	Disabled	68.07	0.01167
	1	Disabled	139488.75	Disabled	45671.45	0.01162

latency. In future work, we plan to evaluate the impact of other power-saving schemes such as *Release Assistance Indication* that indicates the network when to release the connection, *Wake-up signals* that instruct the UE when to start decoding the control and data channels and *Early data transmission* that allows the UE to send or receive data during the random-access procedure.

ACKNOWLEDGMENT

Part of this research was funded by the Flemish FWO SBO S004017N IDEAL-IoT (Intelligent DEnse And Long-range IoT networks) project FWO, GOB7915N, "Modeling and control of energy harvesting wireless sensor networks", and by the ICON project MAGICIAN. MAGICIAN is a project realized in collaboration with imec, with project support from VLAIO (Flanders Innovation and Entrepreneurship). Project partners are imec, Orange, Televic, Citymesh, and REstore.

REFERENCES

[1] *Growth forecast report*. Accessed on 24 July 2019. [Online]. Available: https://www.marketwatch.com/press-release/at-123-cagr-cellular-iot-market-size-is-expected-to-exhibit-us-5110-million-by-2025-2019-06-04.

[2] Y. E. Wang, X. Lin, A. Adhikary, A. Grovlen, Y. Sui, Y. Blankenship, J. Bergman and H. S. Razaghi, "A Primer on 3GPP Narrowband Internet of Things," *IEEE Communications Magazine*, vol. 55, no. 3, pp. 117-123, 2017.

[3] P. A. Maldonado, P. Ameigeiras, J. P. Garzon, J. N. Ortiz and J. M. L. Soler, "NarrowBand IoT Data Transmission Procedures for Massive Machine Type Communications," *IEEE Networks Magazine*, vol. 31, pp. 8-15, 2017.

[4] P.A. Maldonado, M. Lauridsen, P. Ameigeiras, and J.M. Lopez-Soler, "Analytical Modeling and Experimental Validation of NB-IoT Device Energy Consumption," *IEEE Internet of Things Journal*, vol. 6, no. 3, 2019, pp. 5691-5701.

[5] J. Lee and J. Lee, "Prediction-based energy saving mechanism in 3GPP NB-IoT networks," *Sensors (Basel)*, vol. 17, no. 9, 2017.

[6] H. Bello, X. Jian, Y. Wei, and M. Chen, "Energy-Delay Evaluation and optimisation for NB-IoT PSM With Periodic Uplink Reporting," *IEEE Access 2019*, vol. 7, pp. 3074-3081.

[7] A.K. Sultania, P. Zand, C. Blondia, and J. Famaey, "Energy Modeling and Evaluation of NB-IoT with PSM and eDRX," *IEEE Globecom Workshops (GCWkshps)*, Abu Dhabi, UAE, Dec 2018, pp. 1-7.

[8] K. Liu, G. Cui, Q. Li, S. Zhang, W. Wang, and X. Li, "An Optimal PSM Duration Dalculation Algorithm For NB-IoT," *2019 IEEE 5th International Conference on Computer and Communications (ICCC)*, Chengdu, China, Dec. 2019, pp. 447-453.

[9] R. Harwahyu, R.G. Cheng, C.H. Wei and R.F. Sari, "optimisation of Random Access Channel in NB-IoT," *IEEE Internet of Things Journal*, Vol 5, no. 1, Feb 2018, pp 391-402.

[10] A. Azari, C. Stefanovic, P. Popovski and C. Cavdar, "On the Latency-Energy Performance of NB-IoT Systems in Providing Wide-Area IoT Connectivity," *IEEE Transactions on Green Communications and Networking*, 2019, pp. 1-22.

[11] H. Li, G. Chen, Y. Wang and W. Dong, "Accurate Performance Modeling of Uplink Transmission in NB-IoT," *IEEE 24th International Conference on Parallel and Distributed Systems*, 2018, pp. 910-917.

[12] S.-M. Oh, K.-R. Jung, M. S. Bae, and J. Shin, "Performance analysis

for the battery consumption of the 3GPP NB-IoT device," *in Proc. ICTC*, Oct. 2017, pp. 981-983.

[13] S. Xu, Y. Liu, and W. Zhang, "Grouping-based discontinuous reception for massive narrowband Internet of Things systems," *IEEE Internet Things Journal*, vol. 5, no. 3, Jun 2018, pp. 1561-1571.

[14] C. C. Tseng, H. C. Wang, F. C. Kuo, K. C. Ting, H. H. Chen and G. Y. Chen, "Delay and Power Consumption in LTE/LTE-A DRX Mechanism with Mixed Short and Long Cycles," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 3, 2016, pp. 1721-1734.

[15] Y. Miao, W. Li, D. Tian, M. S. Hossain, and M. F. Alhamid, "Narrow band Internet of Things: Simulation and modelling," *IEEE Internet Things Journal*., vol. 5, no. 4, Aug 2018, pp. 2304-2314.

[16] M. E. Soussi, P. Zand, F. Pasveer, and G. Dolmans, "Evaluating the performance of eMTC and NB-IoT for smart city applications," *IEEE Int. Conf. Commun. (ICC)*, Kansas City, MO, USA, May 2018, pp. 1-7.

[17] A.K. Sultania, C. Delgado and J. Famaey, "Implementation of NB-IoT Power Saving Schemes in ns-3",*Workshop on Next-Generation Wireless with ns-3*, pp. 5-8, 2019.

[18] M. Lauridsen, R. Krigslund, M. Rohr, and G. Madueno, "An empirical NB-IoT power consumption model for battery lifetime estimation," *in Proc. IEEE 87th Veh. Technol. Conf. (VTC Spring), Porto, Portugal*, Jun. 2018, pp. 1-5.

[19] *Solution for Cellular IoT*. Accessed = Sept 16, 2020. [Online]. Available: http://www.effnet.com/solutions/cellular_iot/.

[20] *Narrowband Internet of Things Whitepaper, Rohde-Schwarz*. Accessed = Sept 17, 2020. [Online]. Available: https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_application/application_notes/1ma266/1MA266_0e_NB_IoT.pdf.

[21] Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures, v16.2.0, 3GPP Standard TS 36.213, 2020.

[22] Cellular System Support for Ultra Low Complexity and Low Throughput Internet of Things, document 3GPP TR 45.820 v13.0.0, 2015.

[23] H. Malik, H. Pervaiz, M. M. Alam, Y. L. Moullec, A. Kuusik, and M. A. Imran, "Radio resource management scheme in NB-IoT systems," *IEEE Access*, vol. 6, pp. 15051–15064, 2018.

[24] D.L. González, "Theoretical and experimental analysis of NB-IoT technology," Bachelors Degree thesis, *Polytechnic University of Catalonia*, 2019.

[25] M. El Soussi, P. Zand, F. Pasveer and G. Dolmans, "Evaluating the Performance of eMTC and NB-IoT for Smart City Applications", *IEEE International Conference on Communications (ICC)*, 2018.

[26] *Cisco, 2018,Power Saving Mode (PSM) in UEs*. Accessed = May 24, 2019. [Online]. Available: https://www.cisco.com/c/en/us/td/docs/wireless/asr_5000/21/MME/b_21_MME_Admin/b_21_MME_Admin_chapter_0111010.pdf.

[27] F. Xie, 2018, *Power Consumption optimisation.* Accessed = May 24, 2019. [Online]. Available: https://cdn.rohde-schwarz.com/fr/general_37/local_webpages/2018_IoT_test_Day-Power_consumption_Optimisation.pdf.

[28] Martínez, B., F. Adelantado, A. Bartoli, and X. Vilajosana. "Exploring the Performance Boundaries of NB-IoT," *Networking and Internet Architecture, arXiv*, 2018.

[29] *SARA-N2 series*. Accessed = Aug 13, 2020. [Online]. Available: https://www.u-blox.com/en/docs/UBX-18066692.

Ashish Kumar Sultania received the M.Sc. degree in Computer Science from the University of Tartu, Estonia, and Norges teknisk-naturvitenskapelige universitet, Norway in 2017 and B.E. in Information Technology from University of Delhi, India in 2011. He is currently a PhD researcher with the University of Antwerp and imec, Belgium. His research focuses on optimizing energy consumption of IoT devices and their networks. Prior to starting his masters, he worked as a Senior Software Engineer at NXP Semiconductor, India (2011-2015).

Chris Blondia obtained his Master in Science and Ph.D. in Mathematics, both from the Ghent University (Belgium) in 1977 and 1982 respectively. In 1983, he joined Philips Research Laboratory Belgium (PRLB), where he was a researcher between 1986 and 1991 in the group Computer and Communication Systems. Between August 1991 and end 1994 he was an Associate Professor in the Computer Science Department of the University of Nijmegen (The Netherlands). In 1995, he joined the Department of Mathematics and Computer Science of the University of Antwerp, where he is currently a Full Professor in the Internet and Data Lab (IDLab) research group. He has been the Chair of the Department of Mathematics and Computer Science between 2010 and 2016. He is lecturing networking courses. His main research interests are related to the design, analysis, and implementation of algorithms and protocols for communication networks, in particular, wireless networks, focusing on their performance. He has published over 250 papers in international journals and conferences on these research areas. He has been involved in many national and European Research Programs.

Jeroen Famaey is an assistant professor associated with imec and the University of Antwerp, Belgium. He received his M.Sc. degree in Computer Science from Ghent University, Belgium in 2007 and a Ph.D. in Computer Science Engineering from the same university in 2012. He is co-author of over 120 articles published in international peer-reviewed journals and conference proceedings, and 10 submitted patent applications. His research focuses on performance modeling and optimization of wireless networks, with a specific interest in low-power, dense and heterogeneous networks.